In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institute shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the Dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

3/17/65
b

SEQUENCING TO RESOLVE NONSIMULTANEITY CONSTRAINTS IN PROJECT NETWORKS

A THESIS

Presented to

The Faculty of the Graduate Division

by

Venugopal V

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Industrial Engineering

Georgia Institute of Technology

June, 1968

SEQUENCING TO RESOLVE NONSIMULTANEITY

CONSTRAINTS IN PROJECT NETWORKS

Approved:

Chairman

Date approved by Chairman: 5/28/68

ACKNOWLEDGMENTS

I am indebted to Dr. J. Gordon Davis, my thesis advisor, for his inspiring friendship, willing advice, and encouragement, not only during all phases of this research, but throughout the entire duration of my graduate work at this Institute. I am especially grateful to him for new perspective in dealing with portions of the research. I would also like to express my gratitude to Dr. Philip Adler, Jr., and Dr. Lynwood A. Johnson, the members of the reading committee, for their patient examination of the manuscript and an over-all review.

I extend my appreciation to Mrs. Betty Sims for her assistance in typing this thesis and also to Mrs. Imogene Herbert for her generosity in typing the worksheets which appear here.

This thesis is dedicated to my parents, Mr. and Mrs. V. K. Eradi, for their foresight, guidance, and unselfish sacrifice without which the author's graduate program would not have become a reality. I also extend my deepest gratitude to my sister Padmini who always encouraged me toward higher goals in life, and to Mani, whose love, patience and understanding were a constant source of inspiration during the formulation of this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

Figure		Page

SUMMARY

The network-based project management systems have found wide acceptance as a theoretical discipline and application as a practical tool. Network models of projects may not adequately depict true pictures of the projects under consideration if the nonsimultaneity constraint and the precedence constraint are not distinctly recognized. The nonsimultaneity constraint is the restriction imposed on a set of activities that are not precedence-related such that these activities cannot all be in progress concurrently. The inadequacy of a network model can cause considerable losses of effectiveness in project planning, scheduling and control. The purpose of this research is to make explicit the nonsimultaneity constraint and to devise network-based procedures for including the nonsimultaneity constraint into the project scheduling function. Specific procedures are presented by which the activities of a single nonsimultaneous set may be optimally sequenced.

The arbitrary selection of a sequence of the activities within the nonsimultaneous set cannot be credited with having a high probability of resulting in an optimal schedule. However, the examination of every possible network design emanating from the possibility of different sequences within the nonsimultaneous set becomes computationally prohibitive. The research discussed in this paper provides a method for determining the optimal sequence of the activities in the nonsimultaneous set without the necessity of constructing separate networks for each possible sequence within the nonsimultaneous set.

The fundamental nature of the method presented in this research is to consider the basic network as the starting point and to resolve the nonsimultaneity existing among a set of activities by imposing a pseudo-precedence relationship such that this imposed ordering minimizes the increase in project length resulting from such an ordering. A worksheet approach is presented to optimally sequence the three activities in a single nonsimultaneous set of simultaneity maximum of one or two. This approach is extended to deal with the four activities in a single nonsimultaneous set with a simultaneity maximum equal to one. Example applications are furnished to illustrate the implementation of these approaches. Savings in computational magnitude as compared with the construction of separate networks and subsequent calculations on them are indicated to be considerable.

Modified versions of the worksheet approach are presented. These make use of the *boundary sum* values of the activities constituting the nonsimultaneous set. The modified approach does not always result in choosing an optimal sequence. However, they appear to indicate a *good* sequence and the computations are greatly simplified.

A brief discussion about the relationship between the *leveling* and the nonsimultaneity constraint is included. This emphasizes the reasons for leveling and the problems encountered in leveling. It spotlights the necessity of considering both the leveling aspects and the nonsimultaneity constraint as interrelated. Two approaches are suggested to deal with these together.

Recommendations are made for further investigation of other interesting aspects of the nonsimultaneity problem. The extension of the approaches presented in this research to cover the situations involving multiple nonsimultaneous sets and intersecting nonsimultaneous sets is suggested. The design of a computer program to form sequences and evaluate them is recommended. It is suggested to study the possibility of adaptation of the nonsimultaneity problem aspects to the job-shop situation. Suggestions are made relative to the implementation of the sampling procedures to choose a small fraction of all possible sequences. The result might be a procedure which enables us to express the probability of the selected sample containing an optimal sequence.

Solutions for the more basic forms of the problem are shown to be very much simpler than with the computations associated with the design and evaluation of separate networks, one each for a sequence in the nonsimultaneous set.

GLOSSARY OF TERMS AND ABBREVIATIONS

Basic Network	Network which ignores nonsimultaneity constraint.
Modified Network	Network in which nonsimultaneity is assured by the use of an imposed precedence relationship among the activities of the nonsimultaneous set.
Precedence-Related	(Activities) connected in a path of "immediately precedes" relationships, thereby precluding the simultaneity of the activities involved.
Forward Pass	Procedure used to calculate ES and EC.
Backward Pass	Procedure used to calculate LS and LC.
ES	Earliest Activity Start Time.
EC	Earliest Activity Complete Time.
LS	Latest Activity Start Time.
LC	Latest Activity Complete Time.
Total Slack	(LC - EC) = (LS - ES); the extent to which activity may be delayed past its ES without causing project completion to exceed its latest allowable time.
BS	Boundary Sum which is the sum of ES and LC for an activity.
BS'	The term BS calculated on a modified network.
R_x	Resource required for an activity x to be in progress.
R_{MAX}	Maximum resource availability.

CHAPTER I

INTRODUCTION

Purpose

The fundamental purpose of this research is to supplement the potentialities of network-based project management systems by increasing the efficiency with which the nonsimultaneity constraint can be considered when such systems are being used. It is the author's hope that the efforts made in this research will not only lead to improvement in the planning, scheduling and controlling phases of project management, but also open up more avenues for further research. The objective of this research is to design network-based procedures for including the non-simultaneity constraint into the project scheduling function.

Basic Characteristics of Network-Based Project Management

Within the last decade, added to the growing assortment of quantitative tools for business decision making is the Critical Path Method (CPM)--a powerful but fundamentally simple technique for planning, scheduling and controlling large, complex projects. The basic approach involves network representation of projects and has been most closely associated with the names CPM (Critical Path Method) and PERT (Program Evaluation and Review Technique). Though many variations of the basic approach do exist, it is considered apt at this point to describe only the general network technique basic to all these methods.

Project may be defined as a set of interrelated activities required to attain a specified objective by a given method. Large-scale research and development programs, construction work, industrial maintenance and installation operations, and even the production of motion pictures and the planning of hearttransplantation can be cited as a few examples of a project, by this definition. Each of these projects has several characteristics that are essential for analysis by CPM. These are as follows: a) The project consists of a well-defined collection of jobs (or activities) which, when completed mark the end of the project. b) The jobs may be started and stopped independently of each other, within a given sequence. c) The jobs are ordered--that is, they must be performed in technological sequence. For example, excavation must be completed before foundation of a house can be laid. If the goal of one activity must be attained before the next one can commence, the activities are precedence related. Though there is no definite requirement that the activities of a project must be precedence-related, a typical project involves a considerable degree of precedence relationship among its activities.

Network Representation of a Project

The basis of both CPM and PERT is the project network diagram. The network is essentially an outgrowth of the Gantt or bar chart, which is primarily designed to control the time element of a program. The bar chart portrays the major activities forming the program, their scheduled start and finish times, and their current status. The important differences in the network concept are that a) the dependencies of the activities are noted rather explicitly, and b) more detailed definition of activities tends to be made, and c) the graphic representation of time is not required, although it may be used. For simple projects, one can incorporate the dependencies among activities in a traditional bar-chart implicitly. However, for the enormous size of many present-day programs and the tremendous increase in the complexities involved, bar-charts are grossly inadequate as models which can assist management.

For the most part, network-based methods have concentrated on the time parameter and to a somewhat lesser extent on the cost parameter. The performance parameter has not yet been treated explicitly in the network considerations, and this is a much more difficult problem than the other two parameters mentioned since a technical judgment and a valid criterion are required to assess performance and express it in quantitative terms.

Two systems of networking which are most widely used are the Activities-on-Arrows system and the Activities-on-Node system. While the basic concepts underlying both these systems are essentially the same, significant differences exist in the mode of representation. In the former system, the Activities-on-Arrows (A-O-A) system, each job in the project is defined and represented by an appropriately labeled arrow. Each arrow is placed in the network in proper relation to the other arrows. That is, jobs can occur in relation to other jobs in three ways: one job can precede another; jobs can be done concurrently; or one job can follow another. At certain points in the diagram, dotted arrows may be inserted to indicate restraints. These dotted arrows are termed dummies; they do not represent actual jobs, but are put in the diagram to complete the technological work sequence by indicating the interrelationship of one job to another. Dummies are also used to prevent arrows from having common beginning and end points. Table 1 lists some diagrams, which usually appear again and again in networks, together with the logical arrangement they define among operations (27).

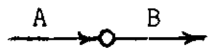
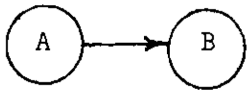
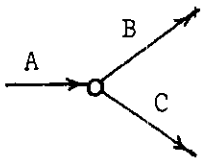
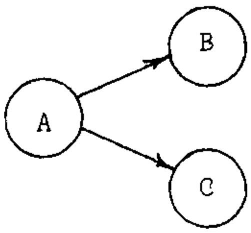
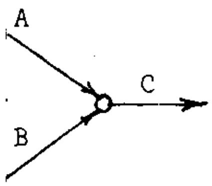
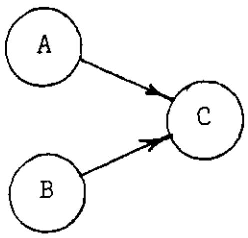
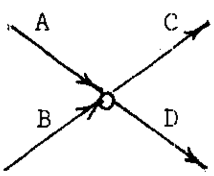
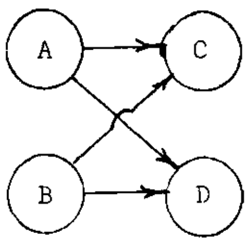
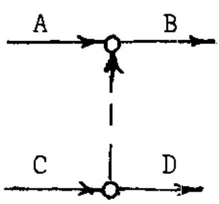
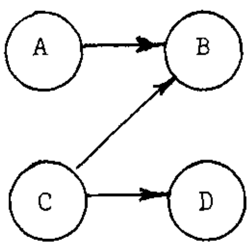
In the Activities-on-Nodes (A-O-N) system, the nodes or circles are used to represent the operations in the project, that is, each circle in the project network corresponds to an operation in the project. Arrows connecting circles are used to depict the relationships among the operations. The A-O-N system was selected as being best suited to the presentation of the research discussed herein. Table 1 gives an idea as to how the same situation is graphically represented in both the systems mentioned. An examination of Table 1 reveals that the *dummy arrow* that is often needed in arrow notation has no analog in a circle notation network. Any logic restriction can be displayed in circle notation without the use of dummy operations. However, dummies can be used as a convenience in the A-O-N system.

Mathematical Basis of Critical Path Method

The formal mathematical properties of the method can be approached from several points of view. The purpose of this section is to describe it from the point of view of mathematical relations and is based mainly on the work done by Levy, Thompson and Wiest (17).

Let $J = \{A,B,C,\ldots,\}$ be a set of jobs that must be done to complete a project. Let $\gg$ denote a relation between two jobs A and B in

Table 1. Common Network Diagrams

A-O-A Diagram	A-O-N Diagram	Logic
		Operation B can begin only after Operation A is completed.
		Neither Operation B nor C can start before operation A is completed, but B and C can be performed concurrently.
		Operation C can begin only after both operations A and B are completed.
		Neither operation C nor D can begin until both A and B are completed, but C can be started independent of D or vice-versa.
		Operation B cannot begin until both A and C are completed, but D can start after only C is completed. (The dotted arrow is a dummy operation which exists only to maintain the logical relationships among A, B, C and D.)

set J, such that B >> A is defined for some pairs of jobs A and B. This is read "B is an *immediate* predecessor of A" or, equivalently, "A is an *immediate* successor of B." The interpretations of the statement B >> A is that job B must be completed before job A can be started. Any given job can be started if and only if *all* its immediate predecessors have been completed. A project is the set J with the >> relationships.

The set $P_a = \{B|\ B >> A\}$ is the immediate predecessor set of job A. Similarly, the set $S_a = \{B|A >> B\}$ is the immediate successor set of job A. The set P_a is the smallest set of jobs in J that must be completed before A can be started. Similarly, S_a is the smallest subset of jobs in J that cannot be started until job A is completed.

A path in the network consists of a subset of J, say $J_i(i=1,2,\ldots,K)$ for which the following relationship holds: $J_1 >> J_2 >> J_3 >> \ldots >> J_K$. The project cannot be completed if there exists any path of the form $J_1 >> J_2 >> \ldots >> J_1$. Such a path is term a loop or a cycle and a network is acyclic only if it contains no loops.

A > B denotes that A precedes B or equivalently, B succeeds A. It is necessary in this case that A does not *immediately* precede B or equivalently, B does not *immediately* succeed A.

As an illustration of the use of the system of description given above, consider a project J which consists of activities A, B, C, D, E and F. The A-O-N network representation is shown in Figure 1.

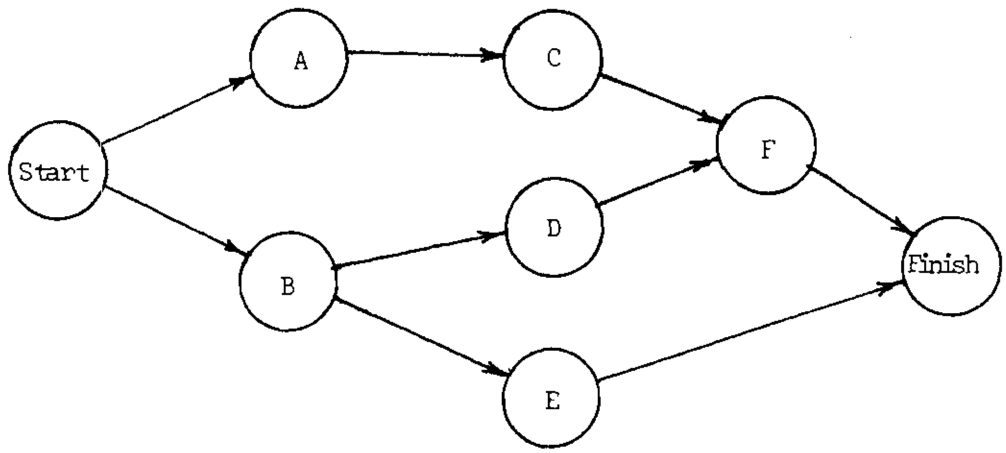


Figure 1. A-O-N Network

START and FINISH are added to the project so that all activities which have no immediate predecessor, are shown to be immediately preceded by START; and also all activities which have no immediate successor, are shown to be immediately succeeded by FINISH. In this illustration, A and B are immediately preceded by START; E and F are immediately succeeded by FINISH. The immediate predecessor and successor sets are: $P_A = \{START\}$, $P_B = \{START\}$, $P_C = \{A\}$, $P_D = \{B\}$, $P_E = \{B\}$, $P_F = \{C,D\}$, $P_{FINISH} = \{E,F\}$; $S_{START} = \{A,B\}$, $S_A = \{C\}$, $S_B = \{D,E\}$, $S_C = \{F\}$, $S_D = \{F\}$, $S_E = \{FINISH\}$, $S_F = \{FINISH\}$.

The following relationships hold: START >> A, START >> B, A >> C, B >> D, B >> E, C >> F, D >> F, E >> FINISH, F >> FINISH: START > C, START > D, START > E, START > F, START > FINISH, A > F, A > FINISH, B > E, B > FINISH, C > FINISH, D > FINISH. The paths from START to FINISH are: START >> A >> C >> F >> FINISH, START >> B >> D >> F >> FINISH, START >> B >> E >> FINISH.

Basic Scheduling Computations

The estimate of the activity mean duration time may be based on a single value which is basically the CPM procedure, or the duration may be based on a system of three time estimates as with the basic PERT approach. Let the duration time of activity A be denoted by t_A. Regardless of which estimation procedure is used, the scheduling computations being described are the same, since they deal only with the estimates of the mean activity duration time.

The basic scheduling computations first involve a forward and a backward pass through the network. Based on a specified occurrence time of the commencement of the project, the forward pass computation gives the earliest start, ES and earliest completion time, EC for each activity in the network. The initial network activity is usually START and the ES of START is assumed to be ZERO, unless otherwise indicated. Suppose now that the project proceeds and every activity in the project is started as soon as all of its immediate predecessors are finished. It is then possible to compute ES for each activity in the project and also EC for each activity.

Let a be any activity such that the early completion EC times of all activities in P_a have already been computed. Then it is possible to compute

$$ES_a = \max_{x \text{ in } P_a} EC(x) \qquad (1)$$

and also

$$EC_a = ES_a + t_a \tag{2}$$

Eventually the early completion time EC of the final activity which is generally taken as FINISH, for convenience, will be computed.

Projects usually have due dates or target dates by which they must be completed. The only target dates that can be met will satisfy the relationship, TARGET DATE ≥ Earliest completion time of the FINISH. If we know a target date, then working backwards from the end of the project, we can compute the latest time at which each job in the project can be completed in order not to delay the entire project beyond the target date. We call this the latest completion (LC) time of the activities. From this, it is also possible to deduce a latest start (LS) time for each activity.

We can define the LC of FINISH to be the target date for completion of the project. If there is no specified target date, the earliest completion EC time of FINISH obtained from the forward pass computations, may be considered as the target date.

Let a be any activity such that the LS times of all activities in the successor set S_a have already been computed. Then we can compute

$$LC_a = \min_{x \text{ in } S_a} LS(x) \tag{3}$$

and also

$$LS_a = LC_a - t_a \tag{4}$$

Eventually, the LS of START will be computed.

Among the many types of slack defined in the literature, two are of most value and are called total activity slack or simply slack, and activity free slack or simply free slack. These are also referred to by some authors as total float and free float.

Total activity slack is equal to the amount of time that an activity completion time can be delayed beyond its EC without affecting the latest completion of the project. This is equivalent to saying that slack is the difference between the earliest and latest allowable Start or Completion times for the activity in question. For any activity a total slack

$$TS_a = LC_a - EC_a = LS_a - ES_a \tag{5}$$

Activity free slack is equal to the amount of time that the activity completion time can be delayed without affecting the earliest start time of any other activity in the network. The equation for calculating the free slack associated with activity a is

$$\text{Free Slack } FS_a = \min_{\text{all } x \text{ in } S_a} (ES_x - EC_a) \tag{6}$$

The criticality of activity a, C_a, is defined (8) as the negative of the slack of a

$$C_a = -TS_a \tag{7}$$

The Critical Path through a network is the path with the least total slack. If we follow the convention of using the EC of the project as the target date or latest completion time of the project, the critical path will have zero slack. Or more specifically, all the activities that lie on the critical path will have zero slack. The path is critical because activities lying along it are the most critical to the attainment of the project completion.

The length of path T_i is represented by $L(T_i)$ and is the sum of the durations of the activities composing the path. Thus, if $J_1 >> J_2 >> \ldots >> J_n$ is path T_1,

$$L(T_1) = \sum_{i=1}^{n} t_{J_i} \tag{8}$$

where t_{J_i} are the activity duration times of J_i, $i = 1,2,\ldots,n$. If $L(T_1)$ is greater than or equal to all other $L(T_j)$ for the network, then T_1 is *the* critical path or one of the critical paths. If latter is the case, it is possible that some activities may be common to more than one critical path.

The Phases of Network-Based Project Management

The purpose of this section is to introduce the reader to the three distinct phases of Project Management. It is claimed that one of the chief advantages of the network approach is that it distinctly recognizes these three phases: Planning, Scheduling and Controlling.

The Planning Phase

No other aspect of project management is so essential to success as planning. The purpose of planning is to establish the end objectives and to define the activities and their interrelationships. This assists in assuring that the project progresses toward the end objectives. A good plan also sets guidelines for corrective action to be taken in case of any unforeseen delay in the progress of the project. CPM planning begins with an analysis of the project objective and a clear definition of the work elements or activities. Making explicit precedence statements may in some cases lead to a division of large work elements into smaller, detailed activities. In other cases, it may be possible to combine a series of work elements into one large activity. The level of detail of the analysis--the level of indenture--depends largely on the purpose of the plan and upon the planner's ability to identify individual activities. The planning stage also involves the determination of the areas of responsibility and authority. Apart from precedence relationships that exist among activities, consideration is given to the feasible time-resource combinations for each activity. It is also possible that, apart from the technological precedence requirements considered, there is a class of activities which, generally, is *external* to the project during planning. This class, most often representing *other constraints*, has a restraining influence on the projected plan. These so-called *other restraints* are of importance in development of project network models. Examples of these restraints are: release of managerial go-ahead decision; availability of funds; availability of suitable weather conditions in an outdoor construction process, etc. Knowledge is also required about the resource limitations like manpower classified into different skills, facilities, money, materials and working space. It is noteworthy that no decision is made during the planning phase as to *when* a particular activity should start.

The Scheduling Phase

Once the planning phase of a project network model has been completed, work can begin on converting the plan into a workable schedule, which can be used as a guide for implementing a project. At the same time, the effect of schedule on two precious company resources--time and money--must be assessed. The logic of a plan can be very appealing, but the timing of its resource requirements can be completely out of phase with the resource availability. The scheduling phase is concerned with establishing starting and completion times for each activity. Since many possible schedules exist even for a small network, it is not feasible to generate and examine all possible schedules. A recommended approach involves generating an all ES schedule and then examine it in light of the restrictions placed on the project. If it is not a satisfactory schedule, it should be altered by applicable techniques, such as resource leveling or the limited resource approach. These approaches may be strengthened by including in them the consideration of alternative activity durations associated with alternative resource levels.

A forward and a backward pass, briefly described in the previous sections, are the basic steps in the scheduling phase. After attaining the earliest start, ES, latest start, LS, earliest completion, EC and the latest completion, LC of each activity in the project, the overall plan can be reviewed to make sure that all the factors regarding technological precedence relationships are considered. Many times, an initial schedule, based on letting all the activities commence at their ES times, may be quite workable from the resource point of view and therefore acceptable if a resource constraint is the only binding factor. However, it may also be possible that the initial schedule as described above does not satisfy all the resource availability restrictions. In such a case, then, it is necessary to delay the start times of one or more of the activities in the project.

This adjustment could result in a lengthening of the project duration in which case there is a definite shift in the focus of attention from the so-called *critical path* and the activities lying on that *path* to the availability of certain key resources. This aspect of project management has received highly inadequate treatment.

The Control Phase

Fundamentally, one is interested in whether a) the project will be completed in time, b) the final cost will be within the estimated amount and, c) the required performance criteria are met. These questions are directed at the project manager, who is responsible for the timely completion of project, meeting required performance standards and staying within the contemplated cost.

In order to control a project, a project manager needs to take corrective action where significant deviations in actual progress and costs from planned progress and costs begin to appear. The basic practical difficulties in taking the necessary remedial measures, stem from the difficulties in measuring the elements of progress and documenting them in such a way as to be able to identify the sources of variation from the schedule. The prerequisite to control is a firm grasp on budgets, costs, schedules and progress, though it also involves the interaction among these elements.

The project manager may also have to revise the initial estimates of the activity duration times and resource requirements, based on the comparison of actual quantities with estimated quantities. The control system should provide adequately for replanning and rescheduling based on the new information available as the project progresses. Lastly, the system should also make provision for the updating of all project documents, if warranted, so that all personnel concerned have the latest schedule in their possession as the project progresses.

The Nonsimultaneity Constraint

The nonsimultaneity constraint is the restriction imposed on a set of activities that are not precedence-related, such that these activities cannot all be in progress concurrently.

If two activities A and B of a network are not bound by any of the precedence relationship $A >> B$, $A > B$, $B >> A$, $B > A$, then this implies that A and B could be performed at the same time. However, this never indicates that A and B *should* be performed simultaneously.

Consider the network on the following page which is a simple A-O-N diagram.

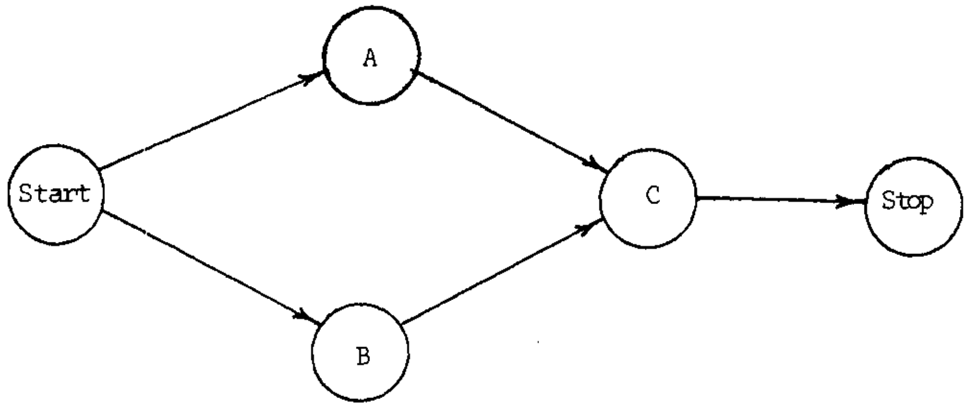


Figure 2. Simple A-O-N Diagram for Illustration of the Nonsimultaneity Constraint

The precedence relationships as implied by the above network are as follows: START >> A, START >> B, START > C, START > STOP; A >> C, A > STOP; B >> C, B > STOP; C >> STOP. There is no >> or > relationship between A and B which implies that A and B could be performed simultaneously.

Now it is possible that some factor other than the technological precedence requirement makes it imperative that A and B cannot be in progress concurrently. This factor gives rise to an additional restriction which could be termed the *nonsimultaneity constraint.* Most common reason for existence of the nonsimultaneity constraint is the restricted availability of key resources. It could also stem from other factors such as those associated with safety. Lack of space in which to work could constitute another reason. Inadequate scope for supervision may give rise to the nonsimultaneity constraint.

The Effect of the Nonsimultaneity Problem on Project Management

It is important to recognize the impact of the nonsimultaneity problem on the different phases of project management. A network should be basically constructed giving consideration to the technological precedence relationships only. If it so happens that the project manager does not distinguish, during the planning stage, the difference between technological precedence relationships and nonsimultaneity constraints, the outcome may be a network which shows constraints which do not actually exist.

Suppose there are two activities A and B which have no precedence relationship of a technological nature whatsoever. If the project manager recognizes that these two activities cannot be performed simultaneously he may be inclined to solve the problem by imposing an additional constraint that B should precede A. Since A >> B is also an equally valid decision to the extent that no technological precedence constraints are violated, it may be possible that A >> B is a better solution than B >> A, when the over-all project is taken into account. It is therefore advisable not to make any decision during the planning phase that will impose a precedence relationship between two activities to resolve a nonsimultaneity. The sequence that is classified intuitively, by the project manager, to be least desirable, may be the one that is actually the optimal or near optimal.

The appropriate stage for the consideration of the nonsimultaneity constraints is the scheduling phase. Thus, equipping the project manager with some means and techniques to resolve the problem of nonsimultaneity during the scheduling phase, will assist him in arriving at a feasible schedule objectively, without having to depend on any of his subjective inclinations. The effect of non-simultaneity constraint on the final schedule will be an imposed precedence relationship.

Some situations may also warrant rescheduling the activities as more information becomes available along with the progress of a project. This need for rescheduling may be as a result of a situation where two activities which were not initially scheduled to be in progress concurrently, are now qualified for simultaneous occurrence due to an unforeseen delay in one of these or both. The schedule that was considered to be optimal at the commencement of the project may not remain optimal throughout the project due to the onset of unforeseen circumstances that crop up from time to time. So it is necessary that all alternative sequences be available for consideration at every rescheduling, if one is to obtain the most satisfactory results. So, in general, procedures that tend to keep the non-simultaneity problem explicit throughout the duration of the project tend to assist the project manager in leading to better overall control of the project.

CHAPTER II

LITERATURE SURVEY

Introduction

It is the purpose of this chapter to review briefly the literature concerning the network approach to project management. Though the review is primarily focussed on the area of resource allocation and the nonsimultaneous constraint aspects of project management, representative literature is cited dealing with the general network-based approaches, because of their possible connection with the problem area under consideration. A brief history of the emergence of project management concepts is included and the reader is also introduced to different versions of the original system. In addition, this chapter is tailored to bring the reader up to date on what has been happening to the Critical Path Method of scheduling since its development and the foreseeable direction it is likely to take.

Brief History of the Emergence of the Project Management Concepts

Though the project idea is very old, it was only in early 1900 that it began to be formally treated. This came about in the process of evolution of the *scientific management* techniques for which Frederick W. Taylor and his contemporaries, of whom Henry L. Gantt was one of the most notable, were primarily responsible.

It was Taylor who recognized the importance of planning and he set up a planning department to assist production. Though Taylor's planning department concept was mainly oriented toward shop management, it was he who recognized the advantages of distinctly separating the *planning* from the operations. He pointed out the fact that the cost of production was lowered by "separating the work of planning and the brainwork as much as possible from the manual labor" (30). It was probably this basic idea that subsequently developed and resulted in the recognition of the importance of distinguishing clearly between the three phases of planning, scheduling and controlling a project. Though Taylor did not publish any literature specifying techniques for this management function, he seems to have been well aware of the problems that could come up when there is a resource constraint. This is evident from what he describes about the various duties of the *balance clerk:*

> The balance clerk should also keep a complete running balance of the hours of work ahead for *each class* of machines and workmen . . . and should keep the manager and sales department posted through daily or weekly condensed reports as to the number of days of work ahead for each department, thus enabling them to obviate either a congestion or scarcity of work.

Gantt (12) apparently was the first to establish the methods for graphic portrayal of different jobs to be planned and scheduled. This methodology, generally known as barchart techniques, helped to achieve systematic planning and scheduling to a considerable extent, and was accepted widely in many industries. Gantt's initial charts were in no way connected with project management and were more closely associated with the "changing and fixing habits of industry" (12), from a physchological point of view. Gantt's contribution to the use of graphics in measurement and control is notable. Several different types of charts had been developed by him and used in the companies with which he had established contact. These finally evolved into the progress chart where the principle involved, or "relating facts to time" was soon applied in many situations as one of the most effective managerial tools (10). Though Gantt's bar-charts were met with acceptance by many, one significant drawback of his approach was that his bar-charts did not portray explicitly the precedence relationship among the activities.

This detriment was partially eliminated when Knoeppel (15) presented a graphic technique which made the precedence relationship explicit. His method, however, did not succeed in gaining appreciable recognition, probably due to the fact that he apparently restricted use of his approach to the filling of production orders. Since Knoeppel was apparently the first to bring the *precedence* into planning and scheduling, he may be considered as the originator of the network representation of a set of interrelated activities.

The next five decades after the time of Knoeppel did not witness any significant improvement in the methodology. This wide gap could have been a result of the fact that those who sought to improve on scheduling techniques took the bar graph as given and it was so ingrained as a part of the thinking of any project manager that he did not make any great effort to overcome the most obvious deficiencies. On the other hand, those specialized in mathematical and logical disciplines did not take much interest in management problems. Then in 1949, line of balance technique was introduced. This resulted in an augmentation of the control features of Knoeppel chart and incorporated an explicit comparison of actual activity progress and the planned progress. The next significant trend toward project orientation and the growth of project organizations within traditional functional organization appeared around 1954. This was chiefly initiated by the urgent need to produce an operational intercontinental ballistic missile in the shortest time. It had become evident that a new and different approach was needed in order to shorten the long lead time usually consumed in producing an operational system. This realization resulted in a tremendous growth in *project activity* and the concept received a burst of attention. As Baumgartner (2) points out, one important factor responsible for this trend of project orientation was the

> . . . rapid technological advance, which resulted from the exceedingly high demands of government projects in terms of capabilities and reaction time, and which dictates that minimum lead time be consumed in developing a system that is not obsolete (although it may be obsolescent) by the time it becomes operational.

The advent of use of electronic computers gave considerable impetus to this development in the project management field. As Muth and Thompson (25) point out, the solution of many problems that existed in the industrial scheduling by hand is impossible. The attempts to make use of high-speed electronic computers in solving them created high motivation to develop new algorithms to cope up with many complex situations.

Network-based Project Management Systems

Early in 1957, Morgan R. Walker of DuPont Engineering Services and James E. Kelley, Jr., then of Remington Rand joined together to explore the possibilities whereby the logic of mathematics could contribute toward a better solution to typical scheculing problems. The outcome of their efforts was their agreement to the effect that a network representation of the job relationships in any project could supply basic information lacking in previous methods. This network concept of depicting a project plan was a bold departure from the traditional bar chart. When the time estimates for each activity were supplied, it became possible to calculate the minimum completion time for a project by simple straightforward rules. By additional simple rules, it became possible to identify which activities were critical and thus define a *critical path* or critical sequence of activities in any project. This approach was the origin of the Critical Path Method.

Once it became possible to state the problem in a network form, Kelley began his further work to solve a more difficult and subtle problem: if a project completion is to be accelerated, which jobs should be expedited and by how much, in order to buy the time advantage at the least cost? This extension of CPM resulted in a *minimum cost expediting--MCX.* Kelley's first solution for MCX was formulated by May 1957 and both CPM and MCX were applied on certain pilot projects at DuPont. Finally in 1958, a plant maintenance shutdown at DuPont left no doubt as to the practical utility and economic value of the method (21).

Kelley is also responsible for an improved MCX solution technique which was a modification of a method published by Ford and Fulkerson in 1955. In 1959, Kelley and Walker, together with John W. Mauchly formed Mauchly Associates, Inc. where other extensions of CPM, such as resources planning and scheduling methods, have been developed.

PERT--Program Evaluation and Review Technique

About June 1958, Willard Fazar of the Special Projects Office of the Navy Bureau of Ordnance, aided by the firm of Booz, Allen and Hamilton, began the development of a network system known as PERT (Program Evaluation and Review Technique). The initial objective of this mission was to plan and coordinate the work of some 3,000 contractors and agencies for the Polaris Missile Program. The schedules of these contractors had to mesh properly and the conflicts had to be seen beforehand and resolved. The harmonious functioning made possible by PERT is credited with advancing the Polaris program more than two years. With PERT, management arbitrarily established the project duration and certain *milestones* within the project had to be met if the project was to finish on time. Another feature of PERT is the estimation of the most probable (normal), latest probable (pessimistic) and the earliest probable (optimistic) activity duration times. With CPM, however, only the normal duration times of the activities are considered and the critical activities are identified at the outset. If management then wants to complete the project in less time than the normal duration, selected jobs are put on a crash basis.

PERT has proved to be a useful tool for monitoring large research and development projects. In its original form, PERT was not particularly intended to be used for cost-control or improving the efficiency of resource-utilization. However, the later extensions of PERT to PERT/COST and other versions bring PERT and CPM closer to form a general network approach, though the philosophies underlying the two techniques were originally quite different.

PERT/COST

PERT/COST is a relatively new system which was designed with the main intent of applying to development programs. This system provides a general operating report for a development program and reflects on the direct relationship between development work and cost and schedule performances.

The term PERT/COST includes the assumption that network must be fully developed before the costing phase can be completed. The basic objectives of PERT/COST are two-fold: a) to achieve a significantly better, or more realistic, original program cost estimate obtained by estimating the cost of each activity in the network; and b) once the program is authorized to proceed, to achieve a marked improvement in control against the original estimate (26).

PERT/RELIABILITY

Frambes (11) and Malcolm (19) discuss PRISM (Program Reliability Information System for Management) and there are two approaches followed under this system. PRISM is being implemented by United States Navy, Operations Research Inc., and Lockheed. PRISM has been used to estimate and monitor the reliability status of the FBM (Fleet Ballistic Missile system). The two approaches to PERT/RELIABILITY are as follows:

a) RPM (Reliability Performance Measure): This approach is directed to the development of a method for predicting quantitatively the ultimate user reliability of the FBM system. RPM provides a prediction in the form of a probabality statement of fraction successful, of the eventual operational reliability of the end item and its subsystem components to be made at each stage of the development cycle.

b) RMI (Reliability Maturity Index): RMI was researched, developed and installed in the Polaris program to help management define a *reliability plan* and to determine how well it is being worked. A reliability event is the start or the completion of an activity resulting in the documentation of a design, a design review, a test, etc. required in the development plan in order to enhance the reliability of the end item. A list of reliability events is made and these events are converted into a time plan showing the start and completion for each documentation required. The network is similar to PERT. RMI provides a running measure of the compliance with planned reliability activities by collecting, analyzing and displaying information on the progress of the reliability documentation program and the quality and significance of the documentation produced (19).

LESS--Least Cost Estimating and Scheduling

This is a system, developed by International Business Machines Corporation, for determining the fastest and most economical method of completing a project using network diagrams. A time-cost slope for each activity and for the project is developed. In real terms, these time-cost slopes represent the additional funding required to shorten the activity per unit time, or alternately, the cost rate of buying time. With the LESS approach, the computer is programmed to systematically reduce the project duration, while minimizing the increase in costs associated with such a reduction in project length (29).

Resource Allocation Methods

There are two basic problems involved in Resource Allocation. One deals with *leveling* the demands for resources while there is a constraint on the total project duration time. The second problem is in the minimization of the total project duration time when there is a constraint on the total availability of certain key resources. The former situation arises when there is no upper bound for the availability of resources but when it is felt desirable to continue the resource utilization at a relatively steady rate during the life of the project. The latter problem occurs when there is a specified limit of available resources and the objective is to schedule the project activities so that the project duration is minimum.

This section is based in part on a survey reported by Edward W. Davis (7). A systematic approach to the problem of resource leveling has been presented by Burgess and Killebrew (5). This approach consists of a method for computing how the total activity level varies throughout the cycle, including a computer program for performing the calculations. Then, a procedure has been presented for rescheduling the activities until the variability in activity level has been reduced to a minimum or near-minimum. The criterion recommended by Burgess and Killebrew is the *sum of the squares* and the objective is to minimize the sum of the squares of resource demands, thereby reducing the variance of the discrete distribution of resource units versus time. Other criteria related to measures of dispersion, like *range,* may also be used. Burgess' algorithm, however, as the authors point out, may not give *the* optimum schedule in all situations.

One computerized manpower-leveling procedure has been presented by L. DeWitte (9). This program enables the individual activities to be scheduled so that manpower fluctuations are minimized while maintaining all the precedence relations. The application of this procedure produces an activity start date which can be used both in direct scheduling and also the forecasting of cost curves. The minimization of the variability is based on an absolute magnitude of deviation from mean level. The specified program can be easily adjusted to other criteria, like least-squaring of fluctuations. Like Burgess' algorithm, DeWitte's procedure is also essentially heuristic in contrast to a linear or dynamic programming approach. The logical trend employed is to split the problem into many subproblems and to further systematically reduce the slack in various activities until all the starting dates are precisely fixed. The resource profile is partitioned into specially-derived intervals and then resources within each interval are sequentially levelled. Computer output may be obtained in histogram form.

A "multiship, multishop, workload-smoothing" program was presented by Levy, Thompson and Wiest (16). This program was designed to level manpower demands in naval shipyards. The approach uses simple rules and applies them probabilistically to optimize manpower requirements. Initially, all jobs are assigned at their earliest possible start times. Then suitable jobs on days which have peak workload are randomly selected. These jobs are moved to later period in time and this shifting is carried out until no further shifting reduces peak loads. Thus, this segment of the program contributes toward smoothing the workloads on all shops concurrently. The next segment performs further leveling on an individual shop basis. However, this program, due to limited amount of application to date, has not been developed yet to suit the analysis of realistic-sized problems.

Wilson (32) presented a slightly different version of the above procedure. The essential difference is in the random choice step. Instead of this, as in the Levy model, Wilson incorporates a dynamic programming scheme at each iteration to arrive at feasible combinations of activities. Another constraint in Wilson's approach appears to be in his assumption of each activity requiring one unit of the same type resource. Thus, the flexibility of the Levy model is reduced in Wilson's version and for problems of considerable size, the procedure becomes computationally prohibitive.

An algorithm for the assembly line balancing problem was given by Gutjahr and Nemhauser (13). The basis of this algorithm was essentially finding a shortest route in a finite directed network. An adaptation of this algorithm for the resource leveling problem was presented by Black (3). This involves generation of possible sets of jobs in the given network using the generated sets as nodes and the predetermined resource constraints as arc lengths. Black's algorithm does take into consideration the situation when more than one resource is involved although for most real-sized project networks, computation is rather complicated. However, this approach and also Wilson's approach are interesting from the conceptual standpoint.

The second basic problem in the area of resource allocation is that of scheduling to satisfy stated resource constraints with an objective to determine the minimum project duration. Burgess' leveling procedure (5) might produce a feasible schedule to satisfy the resource constraints. Kelley (14) offers serial and parallel algorithms for finding a schedule which tends to result in minimum project duration and which remains within the resource constraints governing this period. Kelley also considers the possibility of splitting the activities which compete for resources, whereas this is not so in Burgess' procedure. Kelley also assumes that the activity duration times can be lengthened or shortened with a corresponding reduction or addition of resources required per time unit.

The REST (RESource Time) algorithm (28) offers an added advantage in that it considers the effect of resource level on the performance of the activity. Suppose the most efficient crewsize for a particular activity is 4 and the activity takes 4 days for its completion. Thus, the apparent *work-content* is 16 man-days. However, the situation may be such that 8 men working on the same activity may not be able to complete this activity in 2 days due to certain decrease in efficiency of their performance due to the variation in crew size. This aspect is taken care of, to a certain extent, in the REST algorithm. Along with the possibility of activity splitting, the imposition of a *work interrupt penalty* is also considered in this algorithm.

Another algorithm, though not as flexible as the Kelley routine in regard to the activity splitting and changing their duration times, but which will give good results, obtainable on the first pass, is described by Moder and Phillips. This algorithm is attributed to Brooks (4), now of Auburn University.

Another development in the area of scheduling multi-project activities for the case of stated resource constraint resulted in RAMPS--Resource Allocation and Multi-project Scheduling. This system is a computerized method which is capable of handling several projects simultaneously and scheduling each activity so that the projects are completed by the target dates and an efficient utilization of resources is achieved. The general characteristics of a situation where RAMPS could be implemented may be described as follows: there are one or more projects, each with a desired date for completion. The delay in project completion may be interpreted in dollar values, for example, penalty cost. For each job, the requirement for resources can be described in terms of manpower, machine-power, materials or money. These descriptions include type, combination of resource types, amount of each resource per unit of time, the total amount of each resource required for the completion of the job and the work-interrupt penalty. A measure of efficiency, or lack of it, is also incorporated into the system. The output presentation consists of two types of schedule along with the projected completion date for each project. One type of schedule shows all jobs within each project and the other type indicates all jobs within each resource category (23).

Martino (20) proposes a procedure called MAP--Multiple Resource Allocation Procedure. This utilizes a procedure for assigning priorities to activities competing for limited resources, based on the criterion of *slack*. That is, the eligible activities at a certain point in time are considered in descending order of criticality. In case of a tie, other priority ordering rules are followed in this order until the tie is broken:

(a) Descending order of need of overall number of resource-time units.

(b) Descending order of number of resource units.

(c) Increasing order of sequence code (successor event number).

Moder and Phillips (22) make use of the *Latest Start* criterion for assigning priority in the limited resource problem and achieve the same results as MAP gives, with considerably less computation.

Verhines (31) points out an example where there are two activities competing for resources and assigning priority on the basis of criticality will actually result in longer project duration than the schedule which has the noncritical activity prior to the critical one.

Figure 3 on the following page is the network example cited by Verhines. There is only one repairman available to repair both A and B, and the schedule has to comply with this restraint.

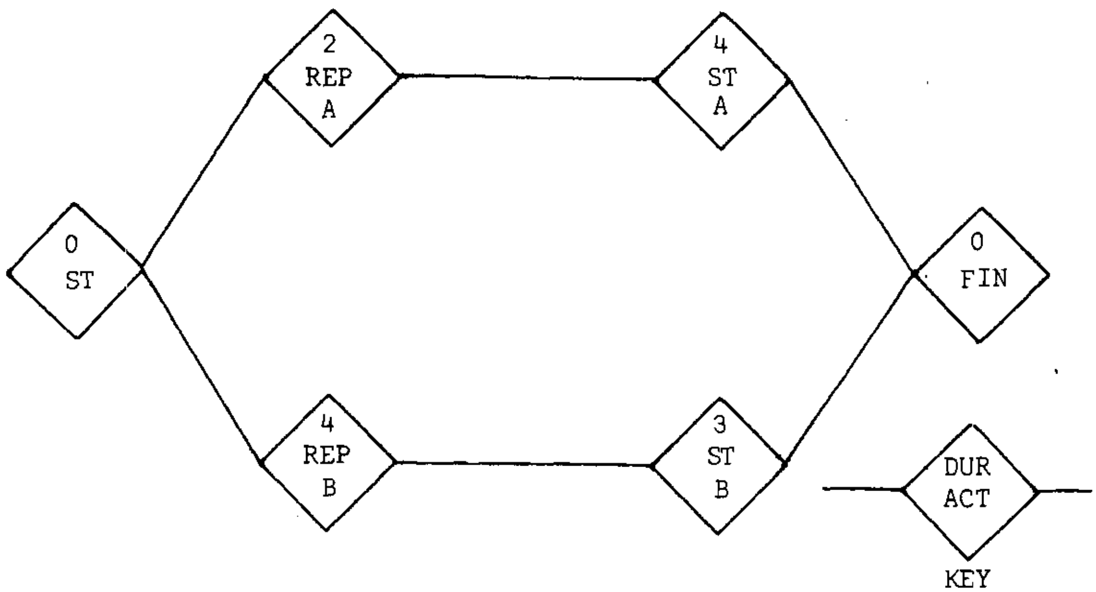


Figure 3. Verhines' Example Project

At time zero, if we assign the priority based on criticality, we will schedule B prior to A since B has more criticality than A. In this case, the project length will be ten time units. However, if A is scheduled prior to B then, the resulting project length will be only nine time units. Thus, Verhines claims that assignment of priority based on criticality alone may not lead to an optimum schedule.

Davis (8) shows that it may be desirable in certain situations not to schedule an activity to start at a given point in the scheduling procedure although all its predecessors are complete, it has scheduling priority and the necessary resources are available.

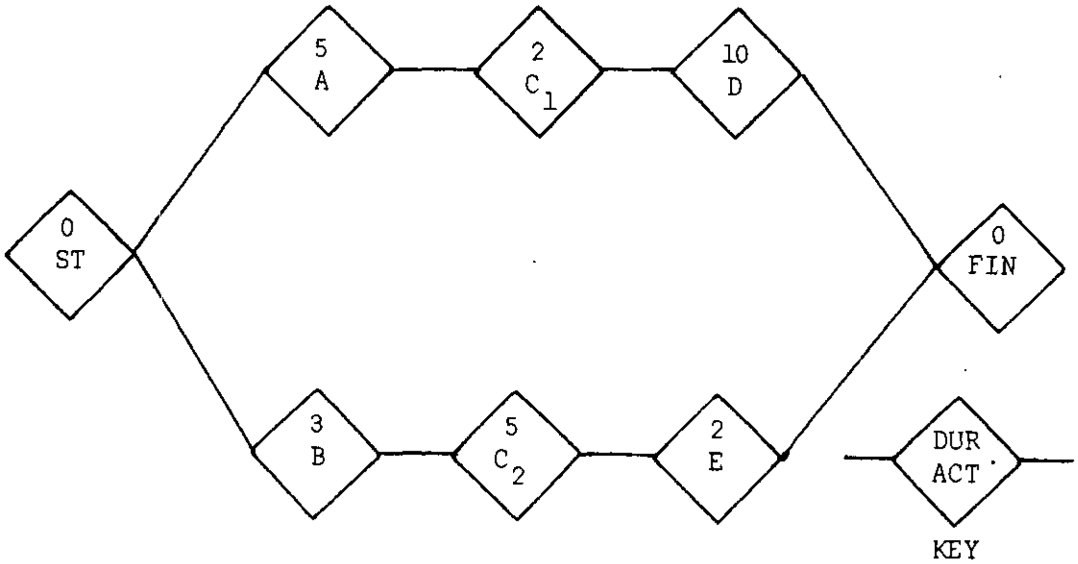


Figure 4. Davis' Example Project

The network in Figure 4 depicts a project in which the activities C_1 and C_2 each require one unit of resource type X and only one unit of resource is available. Most of the previously-mentioned approaches to allocation of resources would stop at time three and examine the unscheduled activities to see which had all their predecessors complete. At this point, C_2 would be the only candidate for starting and since sufficient resources are available, C_2 would be scheduled to start at time three. Assuming that activities cannot be split, starting C_2 at time three means that C_1 can start only at time eight after C_2 is completed, though the activity A, which is the only predecessor to C_1, is completed at time five. The resulting project length is therefore 20 time units. If, however, C_2 is not started until C_1 is completed, a project length of 17 time units is realized. Thus, $C_2 \gg C_1$ would result in a better over-all schedule than would $C_1 >> C_2$, though C_1 is more critical than C_2.

This problem is considered to a limited extent by Shaffer, Ritter and Meyer (27). In their method, RSM (Resource Scheduling Method), they approach the problem of resource allocation by starting with a schedule which has been generated by considering the technological precedence relationships only. Then this schedule is modified by further sequencing of activities so as to resolve the conflict between total resource demand and total resource availability during a given period of time. Thus, the nonfeasibility of the original schedule is eliminated and the resulting schedule becomes workable. The criterion they consider is the increase in project duration caused by the additional sequencing. The objective is to minimize this increase in project duration while resolving the resource conflict. More detailed discussion of RSM is presented in Chapter III. Though RSM lacks the potential of arriving at a multistage optimal solution, it does have an advantage over other methods in that it gives an optimal single-stage solution.

The Nonsimultaneity Constraint

Most of the network methodology developed during the first decade after the evolution of the original network approaches to project manage- do not appear to have incorporated the effect of the nonsimultaneity constraint in the techniques of planning, scheduling and controlling a project. The distinct difference between technological precedence constraint and the nonsimultaneity constraint is either not recognized or not rightly estimated. It was Davis (8) who first treated the problem of nonsimultaneity constraint in network-based project management systems in considerable detail. He treats the scheduling situations when there are multiple as well as single sets of nonsimultaneity activities within a given network. Varying degrees of simultaneity within a set are examined. The basic approach by Davis involves showing all possible sequences of the activities in the nonsimultaneity set in a single network. Then sequences are examined and eliminated by comparing each sequence with the other sequences until the optimal sequence is the only sequence remaining. The project length is considered as the basis to determine the relative *goodness* of the schedule.

Different situations where the nonsimultaneity constraint is involved are discussed. These situations are broadly categorized into three classifications:

(1) The Single Nonsimultaneity set with a simultaneity maximum of one. This depicts a situation where there is a nonsimultaneity set in which none of the activities has any true, or technological precedence requirements imposing partial sequences on the set. The complete enumeration approach involves the generation of a separate network for every possible sequence of the activities in the nonsimultaneity set. Then computation of lengths of the critical paths in each of these networks will obviously identify the optimum sequence. However, this is computationally prohibitive and the enumerative network approach mentioned above reduces the mathematical computation and then simplifies the process of finding out the optimum or near-optimum sequence.

(2) Multiple nonsimultaneity sets when the simultaneity maximum is equal to one. A network may contain more than one nonsimultaneity set. Under this category, the independent and dependent sets are discussed and a generalized procedure for sequencing nonsimultaneity activities, applicable to both dependent and independent sets within the network, is presented. An enumerative network is first constructed. Then a forward pass and a backward pass are made through the enumerative network. When making these passes, some modifications are made in the calculation procedure. *Min Max Criticality* over the sequences in each set are calculated and the set having maximum value of Min Max Criticality is selected. The advantage of single enumerative network approach is found much more striking in the case of multiple nonsimultaneous sets than when a single nonsimultaneity set is involved in the project network.

(3) The nonsimultaneity constraint when the maximum simultaneity is greater than one. The most common nonsimultaneity constraint allows only one activity of a set to be in progress at any one time. However, there may be cases when two or more of the N activities are allowed to be in progress simultaneously. A modified procedure is presented to cope with this situation.

The literature survey indicates the following aspects of network-based project management. A wide variety of analytical and heuristic solutions exist for the time/cost trade-off and the constrained resource allocation problems in the project management. However, none of these takes the nonsimultaneity constraint into consideration. Almost all the scheduling procedures tend to imply the assumption of non-existence of the nonsimultaneity constraint. There is no procedure yet, with proven optimality, which can be universally applicable to all different situations involving the nonsimultaneity constraint. There has not been any criterion established yet which could indicate an optimum sequence from among the many possible combinations of sequences when a nonsimultaneity constraint exists.

CHAPTER III

THE SINGLE NONSIMULTANEOUS SET OF THREE ACTIVITIES WITH SIMULTANEITY MAXIMUM OF ONE OR TWO

Introduction

Of major importance in the analysis of a network is the allocation of resources for the various activities of the project. Any schedule for a project must be one that is feasible from the point of view of resources. If the schedule is one which, at any instant of time, has to draw in more resources than that are available then, it is clearly not feasible. For example, if the plan suggested by a certain network representation of a project requires 14 resource units to be scheduled in the project simultaneously and the available number of resource units is only 7, clearly the CPM plan is of limited usefulness since it leads to schedules which are not practicable.

This chapter presents one procedure for achieving feasibility of a schedule that originally had conflicts due to resource availability. Basically, it is a solution process which utilizes the calculations of the unacceptable plan and the maximum level of resource as then set by the project manager. The solution process resequences the operations such that the previously set level of each resource is not exceeded at any time and such that this resequencing increases the project completion time of the unacceptable CPM plan in a minimal way. Care is taken also to see that the technological precedence requirements of the original network are not violated.

Information Required for Implementation

1. The project network diagram indicating all technological precedence requirements.

2. The duration of each activity and the amount of resources required to perform the activity.

3. The maximum amount of resources that are available. This is a preset level.

It is assumed that the maximum levels of resource availabilities are to be constant for the duration of the project. It is implicit that the maximum available amount of a resource is equal to or greater than the maximum amount of that resource required by any single operation of the project.

The procedure provides a feasible schedule and a new network which is a plan that does not require during any period of time, more of a resource than is originally stated as being available. In other words, the nonsimultaneity constraint involved in the original network is fully considered and the nonfeasibility is completely resolved, giving rise to a new schedule that is workable. More specifically, the modified network will consist of additional pseudo-precedence relationships as required to delay the starts of operations which, if started at their true earliest starts, would require more of a resource than the preset amount.

Before describing the proposed procedure in detail, it is thought appropriate to briefly discuss the Resource Scheduling Method (RSM), presented by Shaffer, Ritter and Meyer in their book (27). They presented a procedure which relieves the resource conflict in a time period by rescheduling the two-operation sequence so that the "Increase in Project Duration (IPD) due to this resolution of resource conflict is minimal." The procedure employed in RSM is the following:

> In any given time period 't', any existing resource conflicts in that time period are relieved by forcing one of the operations involved to follow another operation which requires the resource causing the conflict. The two operations involved in this manipulation are chosen so as to minimize the increase in project duration that results from this sequencing.

For example, consider the A-O-A network shown in Figure 5.

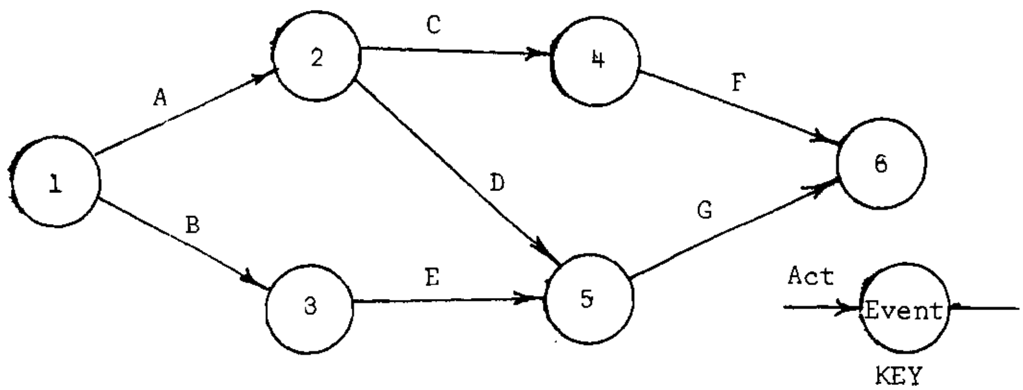


Figure 5. An Example A-O-A Network

Assume that the network as drawn, neglects resource availabilities and requirements. In this network, no precedence relationship exists between operations 2--4 and 3--5, that is, operations C and E.

Assume further that C and E require the same type of resources for their completions. Also assume that the original schedule resulting from the network calculations requires that operations C and E be done simultaneously. However, consider that the doing of C and E simultaneously would require more of a resource than the maximum amount preset by the project manager.

In RSM, this nonsimultaneity constraint is noted. Thus, C and E cannot be scheduled concurrently. RSM calculations show that if E follows C, then the project completion time is increased by a lesser amount than if C were to follow E. As a consequence the RSM calculations state: to relieve the resource conflict between C and E, operation E is to follow C. This condition is incorporated into the network by adding an additional dummy arrow which indicates the additional precedence relationship. It may be noted that RSM deals specifically only with a situation where rescheduling a *two*-operation sequence will resolve the nonsimultaneity problem and make the originally unworkable schedule feasible.

Efforts in this chapter have been directed to achieve an objective of resolving inherent resource conflict in a situation where rescheduling a *three*-operation sequence will resolve the nonsimultaneity constraint.

Suppose A, B, and C are the three activities in a network representation of a project that is subjected to the nonsimultaneity constraint. A, B and C are the three members of the single nonsimultaneous set. Suppose R_A, R_B and R_C are the resource requirements for activities

A, B and C, respectively. We follow a convention of identifying the three activities of the single nonsimultaneous set by A, B C such that

$$R_A \geq R_B \geq R_C$$

This convention is used only for standardizing the worksheet and for convenience in the discussions. Suppose R_{MAX} is the maximum number of resource units available. Then the following relationship holds good.

$$R_A \leq R_{MAX} < R_A + R_B + R_C \quad (9)$$

This is so, since R_{MAX} has to be at least equal to R_A for activity A to be in progress. And if $R_{MAX} \geq R_A + R_B + R_C$, then no nonsimultaneity constraint exists from the point of view of resource availability. The possible combinations of situations where resource conflict could occur in a three-operation sequence are broadly categorized into four separate cases.

CASE I

Case I depicts a situation where

$$R_A + R_B \leq R_{MAX} < R_A + R_B + R_C \quad (10)$$

In this case, it is possible to have *any* two of the three activities scheduled to be in progress concurrently. However, all the three activities cannot be performed simultaneously due to lack of resources.

If R_A, R_B, R_C are 8, 6, 4 resource units, respectively, and if $R_{MAX} = 15$ units, then, this is a situation which comes under this case. It may be noted that

$$(R_A + R_B = 14) \leq (R_{MAX} = 15) < (R_A + R_B + R_C = 18)$$

In this case, it is possible for any of the following sequences to be used to resolve the nonsimultaneity constraint.

a) $A >> B >> C$

b) $A >> C >> B$

c) $B >> A >> C$

d) $B >> C >> A$

e) $C >> A >> B$

f) $C >> B >> A$

g) $A >> \left\{\begin{matrix} B \\ C \end{matrix}\right\}$

h) $B >> \left\{\begin{matrix} C \\ A \end{matrix}\right\}$

i) $C >> \left\{\begin{matrix} A \\ B \end{matrix}\right\}$

j) $\left\{\begin{matrix} B \\ C \end{matrix}\right\} >> A$

k) $\left\{\begin{matrix} C \\ A \end{matrix}\right\} >> B$

l) $\left\{\begin{matrix} A \\ B \end{matrix}\right\} >> C$

$\{{A \atop B}\}$ indicates that A and B *can* be in progress at the same time. It is worth pointing out, at this point, that the increase in project length due to sequencing the activities in the single nonsimultaneous set such that $\{{A \atop B}\} >> C$ will be less than, or at the most, equal to the increase in project length, if the sequencing is $A >> B >> C$. Or, in other words, $\{{A \atop B}\} >> C$ will result in an over-all better schedule, or at least as *good* a schedule as one with $A >> B >> C$ precedence, from the project length considerations. However, $\{{A \atop B}\} >> C$ may not overrule the possibility of $A >> C >> B$ resulting in optimum project length, in certain project configurations. Thus, in case I, an optimum will be one among the sequences (g) through (l) listed above. That is, there will be *an* optimum among the sequences $A >> \{{B \atop C}\}$, $B >> \{{C \atop A}\}$, $C >> \{{A \atop B}\}$, $\{{B \atop C}\} >> A$, $\{{C \atop A}\} >> B$, $\{{A \atop B}\} >> C$.

CASE II

This occurs when

$$R_A + R_C \leq R_{MAX} < R_A + R_B \tag{11}$$

In this case it is possible to have A & C to be in progress concurrently and also B & C but not A and B since $R_{MAX} < R_A + R_B$. If R_A, R_B, R_C are 8, 6, 4 resource units, respectively, and if $R_{MAX} = 12$ units, then this is a situation which comes under case II. It may be noted that

$$(R_A + R_C = 12) \leq (R_{MAX} = 12) < (R_A + R_B = 14)$$

Thus, in this case, it is possible for any of the following sequences to be used to resolve the nonsimultaneity

a) $A >> B >> C$

b) $A >> C >> B$

c) $B >> A >> C$

d) $B >> C >> A$

e) $C >> A >> B$

f) $C >> B >> A$

g) $A >> \{{B \atop C}\}$

h) $B >> \{{C \atop A}\}$

i) $\{{B \atop C}\} >> A$

j) $\{{C \atop A}\} >> B$

The sequences $C >> \{{A \atop B}\}$ and $\{{A \atop B}\} >> C$ are not feasible. Thus, in Case II, *an* optimum will be one among the sequences (g) through (j) listed above, since these will overrule the possibility of one or more of the sequences (a) through (f) being the *only* optimal solutions. That is, there will be *an* optimum among the sequences $A >> \{{B \atop C}\}$, $B >> \{{C \atop A}\}$, $\{{B \atop C}\} >> A$, $\{{C \atop A}\} >> B$.

CASE III

This indicates a situation where

$$R_B + R_C \leq R_{MAX} < R_A + R_C \tag{12}$$

In this case, it is possible to carry out B and C concurrently, but not A and C or A and B simultaneously, since $R_{MAX} < R_A + R_C \leq R_A + R_B$. If R_A, R_B, R_C are 8, 6, 4 resource units, respectively, and if $R_{MAX} = 11$ units, then this is a situation which is an example of Case III.

$$(R_B + R_C = 10) \leq (R_{MAX} = 11) < (R_A + R_C = 12)$$

Thus, in this case, it is possible for any of the following sequences to be feasible:

a) $A >> B >> C$

b) $A >> C >> B$

c) $B >> A >> C$

d) $B >> C >> A$

e) $C >> A >> B$

f) $C >> B >> A$

g) $A >> \{{B \atop C}\}$

h) $\{{B \atop C}\} >> A$

The sequences $B >> \{{C \atop A}\}$, $C >> \{{A \atop B}\}$, $\{{C \atop A}\} >> B$, and $\{{A \atop B}\} >> C$ are not feasible. Now, to find *an* optimum sequence, consideration of sequences $A >> \{{B \atop C}\}$, $\{{B \atop C}\} >> A$ will eliminate the need of computing the increase in project length for the sequences (a), (b), (d) and (f) listed above. However, it is possible that $B >> A >> C$ or $C >> A >> B$ may result in an over-all optimal schedule. That is, there will be *an* optimum among the sequences B >> A >> C, C >> A >> B, A >> $\{{B \atop C}\}$, $\{{B \atop C}\}$ >> A.

CASE IV

Case IV occurs when

$$R_A \leq R_{MAX} < R_B + R_C \tag{13}$$

In this case, it is *not* possible to have *any* combination of activities A, B, and C. If R_A, R_B, R_C are 8, 6, 4 resource units, respectively, and if R_{MAX} = 9 units, then this is a *CASE IV* situation. It may be noted that

$$(R_A = 8) \leq (R_{MAX} = 9) < (R_B + R_C = 10)$$

Thus, in this case, it is possible for any of the following sequences to be practicable:

a) A >> B >> C

b) A >> C >> B

c) B >> A >> C

d) B >> C >> A

e) C >> A >> B

f) C >> B >> A

Sets of $\{{A \atop B}\}$, $\{{B \atop C}\}$, $\{{C \atop A}\}$ are not feasible since

$$R_{MAX} < R_B + R_C \leq R_A + R_C \leq R_A + R_B$$

In other words, Case IV occurs when there is only a single nonsimultaneous set in the project and the simultaneity maximum in this set is equal to one. Thus, all optimal solutions will be revealed by considering the sequences A >> B >> C, A >> C >> B, B >> A >> C, B >> C >> A, C >> A >> B, and C >> B >> A.

The preceding introduces the reader to the problem of the nonsimultaneity constraint and the different cases in which the problem could occur in a single nonsimultaneous set of three activities. The proposed approach to optimally resolve this restraint will now be presented. The sequence that results in the minimum project length is considered as the optimum.

It is evident that if the critical path of the basic network (the network that ignores the nonsimultaneity constraint) extends through one or more activities from a nonsimultaneous set, the sequencing of the activities in that set is likely to affect the length of the project. If the critical path of this basic network does not extend through any of the activities of the nonsimultaneous set, the sequencing of those activities could still cause some or all of the activities to become critical, thus affecting the overall project length. To find out the increase in project length due to the resequencing within the nonsimultaneous set, the only method so far used was to add the additional precedence relationship as suggested by the sequence under consideration, then make a forward pass through the whole network to determine the new project length, thereby finding the increase in project duration. The algorithm presented in this chapter enables one to determine the increase in project length without carrying out the complete forward pass again after pseudo-precedence relationships are added.

To find out the optimal sequence within the single nonsimultaneous set of three activities, it is necessary to determine the sequence which will result in minimum increase in project duration. Enumeration of these IPD's for different possible sequences within the nonsimultaneous set involves the construction of a separate network for each possible sequence within the set and also a forward pass through each of these networks. The proposed approach eliminates the construction of a new network. This approach simplifies the computations considerably, especially if the network is relatively large, involving many activities.

The Proposed Approach

The basic approach involves a forward and backward pass through the basic network. Depending on the maximum availability of resources and the resource requirements of the three activities in question, all the feasible sequences are examined and the IPD for each sequence is calculated without constructing any more networks. The sequence/ sequences having the minimum IPD is/are selected as the optimal sequence/ sequences.

Implementation Procedure

The proposed approach is implemented by using the worksheet No. 1 and following the steps outlined below:

Table 2. Blank Worksheet No. 1

CASE I ☐
CASE II ☐
CASE III ☐
CASE IV ☐

Resource Requirement	Activity	ES	EC	LS	d	Activity Identifier
MAXIMUM =	A					
=	B					
MINIMUM =	C					

TOTAL RESOURCE REQUIREMENTS/AVAILABILITY

Case	I	II	III	IV
	R_{MAX} =			
$R_A + R_B + R_C$ =	>	XXXXXXXXXXXXX		
$R_A + R_B$ =	≤ >	XXXXXXXXX		
$R_A + R_C$ =	XXX ≤	>	XXXX	
$R_B + R_C$ =	XXXXXXXX ≤	>		
R_A =	XXXXXXXXXXXXXX ≤			

LINE		ABC	ACB	BAC	BCA	CAB	CBA	$A >> \{{B \atop C}\}$	$B >> \{{C \atop A}\}$	$C >> \{{A \atop B}\}$	${B \atop C} >> \{A\}$	${C \atop A} >> \{B\}$	$\{{A \atop B}\} >> C$
1	EC	A	A	B	B	C	C	A	B	C	Max^B_C	Max^C_A	Max^A_B
2	LS	B	C	A	C	A	B	Min^B_C	Min^C_A	Min^A_B	A	B	C
3	MAX $\{^{0}_{1-2}$												
4	ES	B	C	A	C	A	B	X X X	X X X	X X X	X X X	X X X	X X X
5	MAX $\{^{1}_{4}$							X X X	X X X	X X X	X X X	X X X	X X X
6	d	B	C	A	C	A	B	X X X	X X X	X X X	X X X	X X X	X X X
7	5 + 6							X X X	X X X	X X X	X X X	X X X	X X X
8	LS	C	B	C	A	B	A	X X X	X X X	X X X	X X X	X X X	X X X
9	MAX $\{^{0}_{7-8}$							0	0	0	0	0	0
10	MAX $\{^{3}_{9}$												

	ABC	ACB	BAC	BCA	CAB	CBA	$A >> \{{B \atop C}\}$	$B >> \{{C \atop A}\}$	$C >> \{{A \atop B}\}$	${B \atop C} >> \{A\}$	${C \atop A} >> \{B\}$	$\{{A \atop B}\} >> C$
CASE IV							XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
CASE III	XXXXXXXXXX	XXXXXXXXXX		XXXXXXXXXX		XXXXXXXXX		XXXXXXXXXX	XXXXXXXXXX		XXXXXXXXXX	XXXXXXXXXX
CASE II	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX			XXXXXXXXX			XXXXXXXXX
CASE I	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX						

Step 1: Identify all members of the nonsimultaneous set and their resource requirements.

Step 2: Fill in these resource requirements under the column *Resource Requirements* such that maximum requirement is for activity A and minimum requirement is for activity C. Thus, we ensure that $R_A \geq R_B \geq R_C$.

Step 3: Record the values of ES, EC, LS and d for these three activities. Note also the activity identifiers in the basic network to ensure correct cross-reference between the A, B, C in the worksheet and the designations of the activities given in the basic network.

Step 4: Record R_{MAX} which is the maximum available number of resource units. Also compute the values for $R_A + R_B + R_C$, $R_A + R_B$, $R_A + R_C$, $R_B + R_C$ and note these in the appropriate place in the work-sheet, on the right-hand top corner of the worksheet. This assists in determining the case under which the particular nonsimultaneous set falls. That is, if

a) $\begin{array}{l} R_A + R_B + R_C > \\ R_A + R_B \quad\quad \leq \end{array} R_{MAX}$ then CASE I occurs

b) $\begin{array}{l} R_A + R_B \quad > \\ R_A + R_C \quad \leq \end{array} R_{MAX}$ then CASE II occurs

c) $\begin{array}{l} R_A + R_C > \\ R_B + R_C \leq \end{array} R_{MAX}$ then CASE III occurs

d) $\begin{array}{l} R_B + R_C > \\ R_A \qquad\quad \leq \end{array} R_{MAX}$ then CASE IV occurs

Check the appropriate square to indicate the Case (I, II, III or IV) on the top left-hand corner of the worksheet.

Step 5: After determining the *case classification* and thus identifying the feasible combinations among which an optimum sequence is bound to lie, we can proceed with the computational steps for calculating IPD with respect to individual feasible sequences.

Step 6: For sequences of form I >> J >> K (for example, A >> B >> C), follow steps 7 through 16. For sequences of form X >> $\{{Y \atop Z}\}$ (for example, A >> $\{{B \atop C}\}$), follow steps 17 through 19. For sequences of form $\{{P \atop Q}\}$ >> R (for example, $\{{B \atop C}\}$ >> A), follow steps 20 through 23.

Step 7: Line one in the worksheet denotes EC_I for sequence IJK. The *I* values are the subscripts for EC and are indicated on the top left corner of each square in this line for all sequences IJK. Thus, the first square in line one is assigned EC_A. Record the value of EC_A in this square. Similarly record values of EC_I for all feasible combinations of IJK.

<u>Step 8</u>: Line two in the worksheet indicates LS_J for sequences IJK. The *J* values or the subscripts of *LS* are given on top left corner in each square. Record the values of LS_J for all feasible sequences IJK.

<u>Step 9</u>: Line three indicates the quantity $\text{MAX}\left\{\begin{matrix}0\\ EC_I - LS_J\end{matrix}\right\}$ for sequences IJK. Compute the value of $EC_I - LS_J$ and record this value if it is greater than zero, otherwise record zero.

<u>Step 10</u>: Line four in the worksheet refers to ES_J values. Record these values.

<u>Step 11</u>: Line five indicates the $\text{MAX}\left\{\begin{matrix}EC_I\\ ES_J\end{matrix}\right\}$. Compare the values of EC_I and ES_J in the same column, that is, those values in lines one and four, and record the greater of the two in this line five.

<u>Step 12</u>: Record d_J values in line six.

<u>Step 13</u>: Under the same column, add the quantities in lines five and six, and enter in line seven. Thus line seven gives the sum $\text{MAX}\left\{\begin{matrix}EC_I\\ ES_J\end{matrix}\right\} + d_J$.

<u>Step 14</u>: Enter LS_K values in line eight.

<u>Step 15</u>: Under each column, subtract the quantity in line eight from that in line seven, and enter the result if nonnegative in line nine. Otherwise, enter zero. Thus, line nine indicates the quantity

$$\text{MAX}\left[\begin{array}{l} \qquad\quad 0 \\ \text{MAX}(\begin{smallmatrix}EC \\ ES_J\end{smallmatrix}I) + d_J - LS_K \end{array}\right. \quad \text{for each IJK sequence.} \tag{14}$$

Step 16: Compare the quantities in line three and line nine and enter the greater of the two in line ten. Thus, in each column of line ten, is the amount

$$\text{MAX}\left[\begin{array}{l} \text{MAX}\ \{\begin{smallmatrix} 0 \\ EC_I - LS_J \end{smallmatrix}\} \\ \\ \text{MAX}\ \{\begin{smallmatrix} 0 \\ \text{MAX}\ (\begin{smallmatrix}EC \\ ES_J\end{smallmatrix}I) + d_J - LS_K \end{smallmatrix} \end{array}\right. \tag{15}$$

and this gives the IPD for each sequence IJK. Go to step 23.

Step 17: For sequences of the form X >> $\{\begin{smallmatrix}Y \\ Z\end{smallmatrix}\}$(for example, A >> $\{\begin{smallmatrix}B \\ C\end{smallmatrix}\}$), record EC_X in line one under the column for the sequence under consideration.

Step 18: For sequences of form X >> $\{\begin{smallmatrix}Y \\ Z\end{smallmatrix}\}$, line two in the worksheet indicates the minimum of LS_Y and LS_Z. Thus, for sequence A >> $\{\begin{smallmatrix}B \\ C\end{smallmatrix}\}$, record the minimum of LS_B and LS_C in line two under the column with sequence heading as A >> $\{\begin{smallmatrix}B \\ C\end{smallmatrix}\}$.

Step 19: In the column for sequence under consideration, subtract quantity in line two from that in line one, and enter the result if non-negative in line three. Otherwise, enter zero. Thus, line three indicates the quantity

$$\mathrm{MAX}\begin{bmatrix} 0 \\ \\ EC_X - \mathrm{MIN}\left({LS_Y \atop LS_Z}\right) \end{bmatrix} \tag{16}$$

for sequence X >> $\{{Y \atop Z}\}$. Re-enter this quantity in the same column in line ten to facilitate comparison of all the IPD's. Go to step 23.

Step 20: For sequences of the form $\{{P \atop Q}\}$ >> R, record $\mathrm{MAX}\left({EC_P \atop EC_Q}\right)$ in line one, in the column for the sequence under consideration. For example, in the column related to $\{{B \atop C}\}$ >> A sequence, line one will have $\mathrm{MAX}\left({EC_B \atop EC_C}\right)$.

Step 21: For sequences of the form $\{{P \atop Q}\}$ >> R, record LS_R in line two in the column for the sequence under consideration. For example, in the column related to $\{{B \atop C}\}$ >> A sequence, line two will have LS_A.

Step 22: In the column for the sequence under consideration, subtract the quantity in line two from that in line one, and if nonnegative record this in line three. Otherwise, record zero. Thus, line three indicates the quantity

$$\mathrm{MAX}\begin{bmatrix} 0 \\ \\ \mathrm{MAX}\left({EC_P \atop EC_Q}\right) - LS_R \end{bmatrix} \quad \text{for the sequence } \{{P \atop Q}\} >> R. \tag{17}$$

Re-enter this quantity in the same column in line ten. This amount is IPD.

Step 23: Having obtained the values of IPD for all feasible combinations, one, some, or all of which are likely to be optimal, select that sequence or those sequences that have the shortest IPD. These are the optimal sequences within the single nonsimultaneous set of three activities in the project. The increase in project duration due to this additional sequence restraint within the nonsimultaneous set is known to be the IPD.

Apart from enabling the project manager to become aware of *the* optimal sequence or sequences, in advance, the worksheet also enables him to know the over-all effect of any resequencing within the single nonsimultaneous set. The algorithm presented is relatively simple to apply in the case of three activities, having any kind of nonsimultaneity. This approach also eliminates the need of making a forward pass and a backward pass on the network which is modified to incorporate the pseudo-precedence relationships which resolve the nonsimultaneity.

Example Application

Step 1: The nonsimultaneous set consists of activities E, F, G and the resource requirements are given to be 8, 6, 4 units, respectively.

Step 2: Record these resource requirements on the worksheet such that maximum requirement (8 units in this example) is for activity A (as in the worksheet) and minimum resource requirement (4 units) is for activity C. Thus, activity A in the worksheet corresponds to activity E in the basic network, B corresponds to F, and C to G.

Step 3: Record the ES, EC, LS and d for the activities. These values are from the basic network.

Step 4: Enter $R_{MAX} = 15$ on top left-hand corner. Also compute

$$R_A + R_B + R_C = 18 \qquad R_{MAX} = 15$$

$$R_A + R_B = 14$$

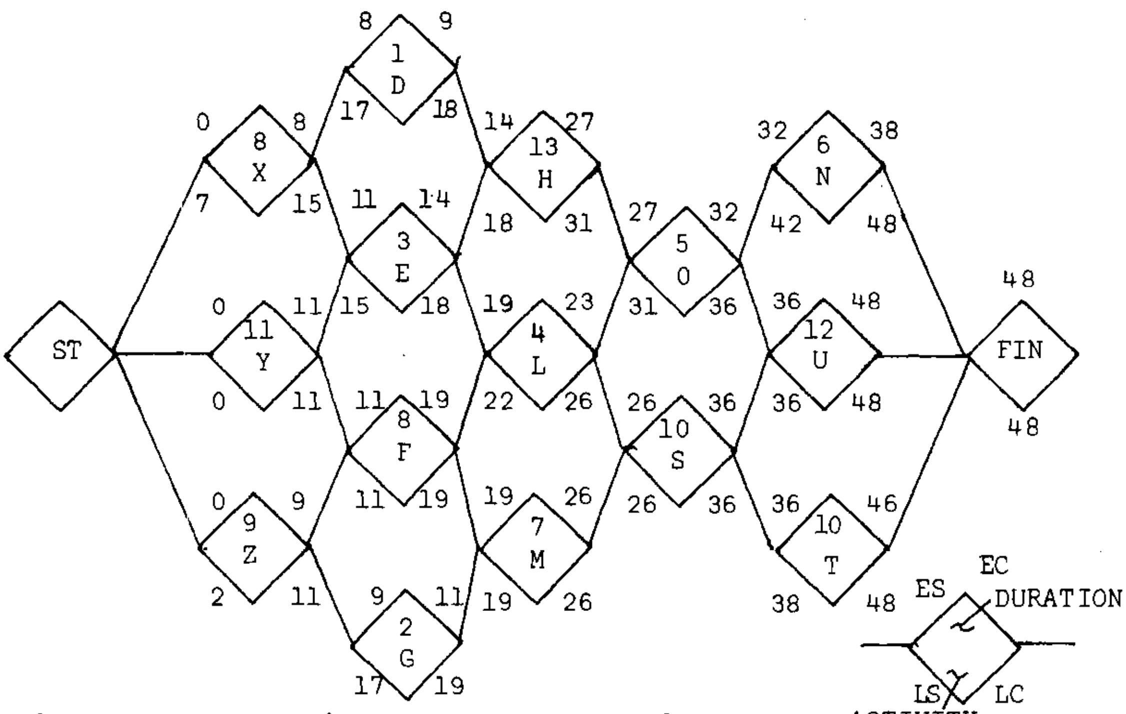
$$R_A + R_C = 12$$

$$R_B + R_C = 10$$

$$R_A = 8$$

Since $R_A + R_B + R_C = 18 > R_{MAX} = 15$, $R_A + R_B = 14 \leq R_{MAX} = 15$, CASE I occurs.

Check the square corresponding to CASE I.



Given: E,F,G require 8,6,4 resource units, respectively. Maximum availability resource = 15 units. E,F G form the single nonsimultaneity set.

Figure 6. Example Basic Network for Application of Worksheet Approach

Table 3. Worksheet No. 1 Used for a Single Nonsimultaneous Set of Three Activities with a Simultaneity Maximum Greater than One

CASE I ☒
CASE II ☐
CASE III ☐
CASE IV ☐

Resource Requirement	Activity	ES	EC	LS	d	Activity Identifier
MAXIMUM =8	A	11	14	15	3	E
=6	B	11	19	11	8	F
MINIMUM =4	C	9	11	17	2	G

TOTAL RESOURCE REQUIREMENTS/AVAILABILITY

Case	I	II	III	IV
	R_{MAX} = 15			
$R_A + R_B + R_C$ = 18 > 15	XXX	XXX	XXX	XXX
$R_A + R_B$ = 14 ≤ 15 >			XXX	XXX
$R_A + R_C$ = 12	XXX ≤		>	XXX
$R_B + R_C$ = 10	XXX ≤		>	
R_A = 8	XXX ≤			

LINE		ABC	ACB	BAC	BCA	CAB	CBA	$A >> \{^B_C\}$	$B >> \{^C_A\}$	$C >> \{^A_B\}$	$\{^B_C\} >> A$	$\{^C_A\} >> B$	$\{^A_B\} >> C$
1	EC	A	A	B	B	C	C	A 14	B 19	C 11	Max^B_C 19	Max^C_A 14	Max^A_B 19
2	LS	B	C	A	C	A	B	Min^B_C 11	Min^C_A 15	Min^A_B 11	A 15	B 11	C 17
3	MAX $\{^0_{1-2}$							3	4	0	4	3	2
4	ES	B	C	A	C	A	B	X X X	X X X	X X X	X X X	X X X	X X X
5	MAX $\{^1_4$							X X X X X X X X	X X X X X X X X	X X X X X X X X	X X X X X X X X	X X X X X X X X	X X X X X X X X
6	d	B	C	A	C	A	B						
7	5 + 6												
8	LS	C	B	C	A	B	A	X X X	X X X	X X X	X X X	X X X	X X X
9	MAX $\{^0_{7-8}$							0	0	0	0	0	0
10	MAX $\{^3_9$							3	4	0	4	3	2

	ABC	ACB	BAC	BCA	CAB	CBA	$A >> \{^B_C\}$	$B >> \{^C_A\}$	$C >> \{^A_B\}$	$\{^B_C\} >> A$	$\{^C_A\} >> B$	$\{^A_B\} >> C$
CASE IV							XXX	XXX	XXX	XXX	XXX	XXX
CASE III	XXX	XXX		XXX		XXX		XXX	XXX		XXX	XXX
CASE II	XXX	XXX	XXX	XXX	XXX	XXX			XXX			XXX
CASE I	XXX	XXX	XXX	XXX	XXX	XXX						

Step 5: Feasible combinations to be considered are indicated in the bottom of the worksheet. Thus, for Case I, the feasible combinations to be considered are

$$A >> \{{}^{B}_{C}\},\ B >> \{{}^{C}_{A}\},\ C >> \{{}^{A}_{B}\}, \{{}^{B}_{C}\} >> A,\ \{{}^{C}_{A}\} >> B, \{{}^{A}_{B}\} >> C$$

Step 6: For sequences $A >> \{{}^{B}_{C}\}$, $B >> \{{}^{C}_{A}\}$, $C >> \{{}^{A}_{B}\}$ follow steps 17 through 19. For sequences $\{{}^{B}_{C}\} >> A$, $\{{}^{C}_{A}\} >> B$, $\{{}^{A}_{B}\} >> C$, follow steps 20 through 23.

Steps 7 Through 16: These steps do not apply in this case.

Step 17: For sequence $A >> \{{}^{B}_{C}\}$, enter $EC_A = 14$ in line one.

For sequence $B >> \{{}^{C}_{A}\}$, enter $EC_B = 19$ and

For sequence $C >> \{{}^{A}_{B}\}$, enter $EC_C = 11$.

Step 18: For sequence $A >> \{{}^{B}_{C}\}$, enter $MIN\{{}^{LS_B}_{LS_C}\} = MIN\{{}^{11}_{17}\} = 11$ in line two.

For sequence $B >> \{{}^{C}_{A}\}$, enter $MIN\{{}^{LS_C}_{LS_A}\} = MIN\{{}^{17}_{15}\} = 15$.

For sequence $C >> \{{}^{A}_{B}\}$, enter $MIN\{{}^{LS_A}_{LS_B}\} = MIN\{{}^{15}_{11}\} = 11$.

Step 19: For sequence $A >> \{{}^{B}_{C}\}$, subtract the quantity in line two from that in line one, that is, $14 - 11 = 3 > 0$ and so enter 3 in line three. For sequence $B >> \{{}^{C}_{A}\}$, we have $19 - 15 = 4 > 0$ and so enter 4 in line three. For sequence $C >> \{{}^{A}_{B}\}$, we have $11 - 11 = 0$ and so enter zero in line three. Enter these nonnegative quantities 3, 4, 0 in line 10 under the columns $A >> \{{}^{B}_{C}\}$, $B >> \{{}^{C}_{A}\}$, $C >> \{{}^{A}_{B}\}$, respectively. Thus, these sequences have IPD's of 3, 4 and 0, respectively.

Step 20: For sequences $\left\{{B \atop C}\right\} >> A$ record $\mathrm{MAX}\left\{{EC_B \atop EC_C}\right\}$ in line one, that is, $\mathrm{MAX}\left\{{19 \atop 11}\right\} = 19$. Similarly for sequences

$$\left\{{C \atop A}\right\} >> B, \text{ record } \mathrm{MAX}\left\{{EC_C \atop EC_A}\right\} = \mathrm{MAX}\left\{{11 \atop 14}\right\} = 14$$

$$\left\{{A \atop B}\right\} >> C, \text{ record } \mathrm{MAX}\left\{{EC_A \atop EC_B}\right\} = \mathrm{MAX}\left\{{14 \atop 19}\right\} = 19$$

Step 21: In line two of the worksheet,

$$\text{For } \left\{{B \atop C}\right\} >> A, \text{ record } LS_A = 15$$

$$\text{For } \left\{{C \atop A}\right\} >> B, \text{ record } LS_B = 11$$

$$\text{For } \left\{{A \atop B}\right\} >> C, \text{ record } LS_C = 17$$

Step 22: For $\left\{{B \atop C}\right\} >> A$, subtract the quantity in line two from that in line one, that is, $19 - 15 = 4 > 0$ and so enter 4 in line three. Similarly, for $\left\{{C \atop A}\right\} >> B$, we have $14 - 11 = 3 > 0$ and so enter 3 in line three; for $\left\{{A \atop B}\right\} >> C$, we have $19 - 17 = 2 > 0$ and so enter 2 in line three. Re-enter these quantities 4, 3, 2 in line ten under the columns for $\left\{{B \atop C}\right\} >> A$, $\left\{{C \atop A}\right\} >> B$, $\left\{{A \atop B}\right\} >> C$, respectively. These sequences thus have their IPD's of 4, 3, 2, respectively.

Step 23: Compare the IPD's obtained for different sequences and select the one which has the minimum IPD. Thus, comparing 3, 4, 0, 4, 3, 2, we see that 0 is MINIMUM IPD and is for sequence $C >> \left\{{A \atop B}\right\}$. Thus, for the example considered, OPTIMUM sequence is $C >> \{{A \atop B}\}$ when R_{MAX} is given to be 15 units.

We will now consider the same example project again and assume that R_{MAX} = 8 units instead of 15 units as before. The purpose of this is to illustrate an example of CASE IV and thus show the computations for sequences ABC, ACB, BAC, BCA, CAB and CBA which have not been illustrated in the previous example. It may be noted that the basic network is the same as that for example one; but R_{MAX} is different.

Example Two

Step 1: The nonsimultaneous set consists of activities E, F, G and the resource requirements are given to be 8, 6, 4 units, respectively.

Step 2: Same as step two in Example 1.

Step 3: Same as step three in Example 1.

Step 4: Record R_{MAX} = 8 on top left-hand corner. Also compute

$$R_A + R_B + R_C = 18 \qquad R_{MAX} = 8$$

$$R_A + R_B = 14$$

$$R_A + R_C = 12$$

$$R_B + R_C = 10$$

$$R_A = 8$$

Since

$$\begin{array}{ll} R_B + R_C = 10 > & \\ & R_{MAX} = 8, \\ R_A = 8 \leq & \end{array}$$

Table 4. Worksheet No. 1 Used for a Single Nonsimultaneous Set of Three Activities when Simultaneity Maximum is Equal to One

CASE I ☐
CASE II ☐
CASE III ☐
CASE IV ☒

Resource Requirement	Activity	ES	EC	LS	d	Activity Identifier
MAXIMUM = 8	A	11	14	15	3	E
= 6	B	11	19	11	8	F
MINIMUM = 4	C	9	11	17	2	G

TOTAL RESOURCE REQUIREMENTS/AVAILABILITY

Case	I	II	III	IV
	$R_{MAX} = 8$			
$R_A + R_B + R_C = 18$	>	XXXXXXXXXXXXXX		
$R_A + R_B = 14$	≤	>	XXXXXXXXXXX	
$R_A + R_C = 12$	XXX ≤	>	XXXX	
$R_B + R_C = 10$	XXXXXXXX ≤	> 8		
$R_A = 8$	XXXXXXXXXXXXXX ≤			

LINE		ABC	ACB	BAC	BCA	CAB	CBA	$A >> \{^B_C\}$	$B >> \{^C_A\}$	$C >> \{^A_B\}$	$\{^B_C\} >> A$	$\{^C_A\} >> B$	$\{^A_B\} >> C$
1	EC	A 14	A 14	B 19	B 19	C 11	C 11	A	B	C	Max^B_C	Max^C_A	Max^A_B
2	LS	B 11	C 17	A 15	C 17	A 15	B 11	Min^B_C	Min^C_A	Min^A_B	A	B	C
3	MAX $\{^{0}_{1-2}$	3	0	4	2	0	0						
4	ES	B 11	C 9	A 11	C 9	A 11	B 11	X X X	X X X	X X X	X X X	X X X	X X X
5	MAX $\{^1_4$	14	14	19	19	11	11	X X X	X X X	X X X	X X X	X X X	X X X
6	d	B 8	C 2	A 3	C 2	A 3	B 8	X X X	X X X	X X X	X X X	X X X	X X X
7	5 + 6	22	16	22	21	14	19	X X X	X X X	X X X	X X X	X X X	X X X
8	LS	C 17	B 11	C 17	A 15	B 11	A 15	X X X	X X X	X X X	X X X	X X X	X X X
9	MAX $\{^{0}_{7-8}$	5	5	5	6	3	4	0	0	0	0	0	0
10	MAX $\{^3_9$	5	5	5	6	3	4						

	ABC	ACB	BAC	BCA	CAB	CBA	$A >> \{^B_C\}$	$B >> \{^C_A\}$	$C >> \{^A_B\}$	$\{^B_C\} >> A$	$\{^C_A\} >> B$	$\{^A_B\} >> C$
CASE IV							XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX
CASE III	XXXXXXXXXX	XXXXXXXXXX		XXXXXXXXXX		XXXXXXXXXX		XXXXXXXXXX	XXXXXXXXXX		XXXXXXXXXX	XXXXXXXXXX
CASE II	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX			XXXXXXXXX			XXXXXXXXX
CASE I	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX	XXXXXXXXXX						

CASE IV occurs. Check the sequence corresponding to CASE IV.

Step 5: Feasible combinations to be considered are:

$A >> B >> C$, $A >> C >> B$, $B >> A >> C$

$B >> C >> A$, $C >> A >> B$, $C >> B >> A$

Step 6: For the above sequences which are of form $I >> J >> K$, follow steps 7 through 16.

Step 7: Enter the EC_I values in line one for all IJK sequences. Thus,

for ABC and ACB sequences, enter $EC_A = 14$

for BAC and CAB sequences, enter $EC_B = 19$

for CAB and CBA sequences, enter $EC_C = 11$

Step 8: Line two in the worksheet indicates LS_J values for sequences IJK. Enter LS_J values in line two. Thus,

for sequence ABC, it is $LS_B = 11$

ACB, it is $LS_C = 17$

BAC, it is $LS_A = 15$

BCA, it is $LS_C = 17$

CAB, it is $LS_A = 15$

CBA, it is $LS_B = 11$

Step 9: Compute the value of $EC_I - LS_J$, that is, subtract quantity in line two from that in line one under each column, and if the quantity is nonnegative, then enter it, otherwise enter zero. Thus

for sequence ABC, $14 - 11 = 3 > 0$; so enter 3

ACB, $14 - 17 = -3 < 0$; so enter 0

BAC, $19 - 15 = 4 > 0$; so enter 4

BCA, $19 - 17 = 2 > 0$; so enter 2

CAB, $11 - 11 = 0$; so enter 0

Step 10: Enter ES_J values. Thus,

for sequence ABC, $ES_B = 11$

ACB, $ES_C = 9$

BAC, $ES_A = 11$

BCA, $ES_C = 9$

CAB, $ES_A = 11$

CBA, $ES_B = 11$

Step 11: Compare values of EC_I and ES_J (that is, the values in line one and line four), for each sequence IJK and enter the greater of the two in line five. Thus,

for sequence ABC, compare $EC_A = 14$ and $ES_B = 11$ and

since 14 is the greater, enter 14 in line five. Similarly,

for ACB, $EC_A = 14$, $ES_C = 9$; enter 14 in line five

BAC, $EC_B = 19$, $ES_A = 11$; enter 19 in line five

BCA, EC_B = 19; ES_C = 9; enter 19 in line five

CAB, EC_C = 11, ES_A = 11; enter 11 in line five

CBA, EC_C = 11, ES_B = 11; enter 11 in line five

Step 12: Enter d_J values in line six. Thus, we have 8, 2, 3, 2, 3, 8 in line six which are the values of d_B, d_C. d_A, d_C, d_A, d_B, respectively.

Step 13: In each column, add quantities in line five and line six and enter in line seven. Thus,

for sequence, ABC, we have 14 + 8 = 22

ACB, we have 14 + 2 = 16

BAC, we have 19 + 3 = 22

BCA, we have 19 + 2 = 21

CAB, we have 11 + 3 = 14

CBA, we have 11 + 8 = 19

Step 14: Enter LS_K values in line eight. Thus,

for sequence ABC, LS_C = 17

ACB, LS_B = 11

BAC, LS_C = 17

BCA, LS_A = 15

CAB, LS_B = 11

CBA, LS_A = 15

Step 15: In each column, subtract the quantity in line eight from that in line seven and if the result is nonnegative, enter it in line nine; otherwise enter zero in line nine. Thus,

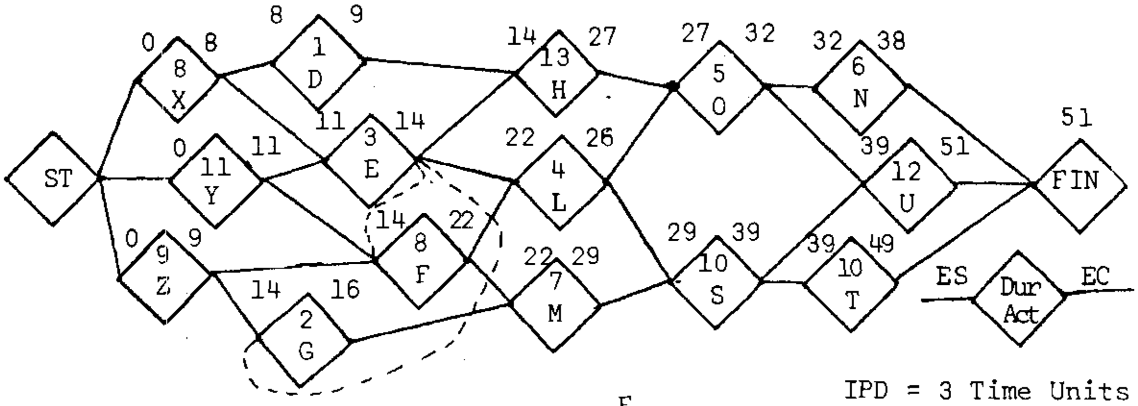
for ABC, 22 - 17 = 5 > 0; so enter 5
ACB, 16 - 11 = 5 > 0; so enter 5
BAC, 22 - 17 = 5 > 0; so enter 5
BCA, 21 - 15 = 6 > 0; so enter 6
CAB, 14 - 11 = 3 > 0; so enter 3
CBA, 19 - 15 = 4 > 0; so enter 4

Step 16: In each column, compare the quantities in line three and line nine and enter the greater of these two in line ten. Thus, for sequence ABC, compare 3 and 5; enter 5 in line ten.

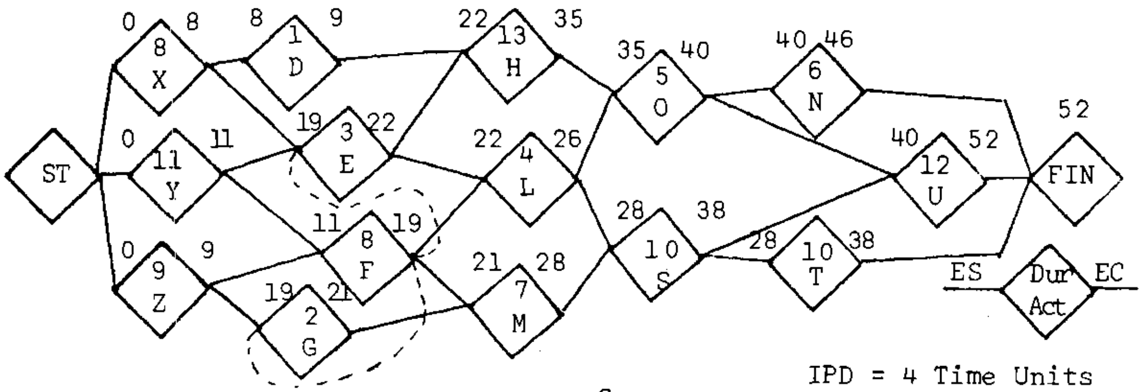
For sequence ACB, we have 0 and 5; enter 5
BAC, we have 4 and 5; enter 5
BCA, we have 2 and 6; enter 6
CAB, we have 0 and 3; enter 3
CBA, we have 0 and 4; enter 4

These are the values of IPD's. Go to step 23.

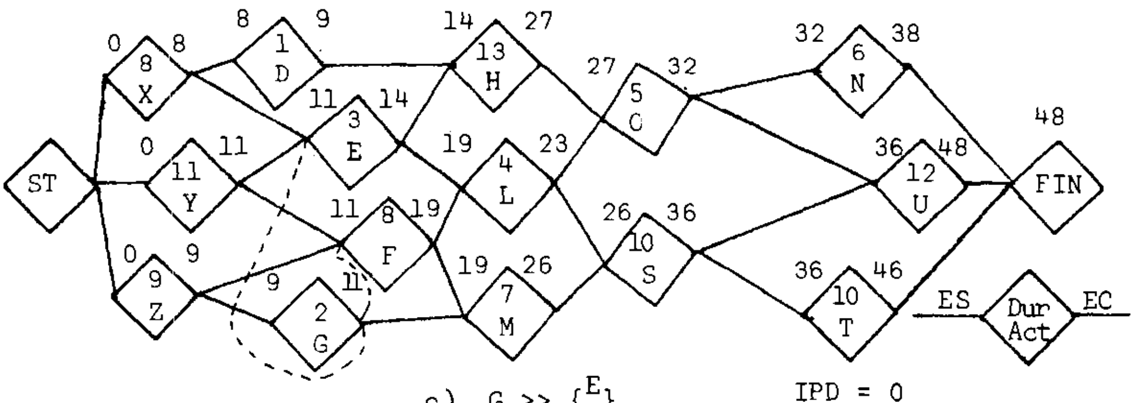
Step 23: Compare the IPD's obtained for different sequences and select the one which has the MINIMUM IPD. Thus, comparing 5, 5, 5, 6, 3 and 4, we see that 3 is Minimum IPD and is for sequence CAB. Thus, for the example considered, the optimum sequence is CAB and the resequencing causes an increase in project duration of three time units from that of



a) E >> $\{^{F}_{G}\}$



b) F >> $\{^{G}_{E}\}$



c) G >> $\{^{E}_{F}\}$

Figure 7. Separate Networks for Different Sequences Within the Nonsimultaneous Set

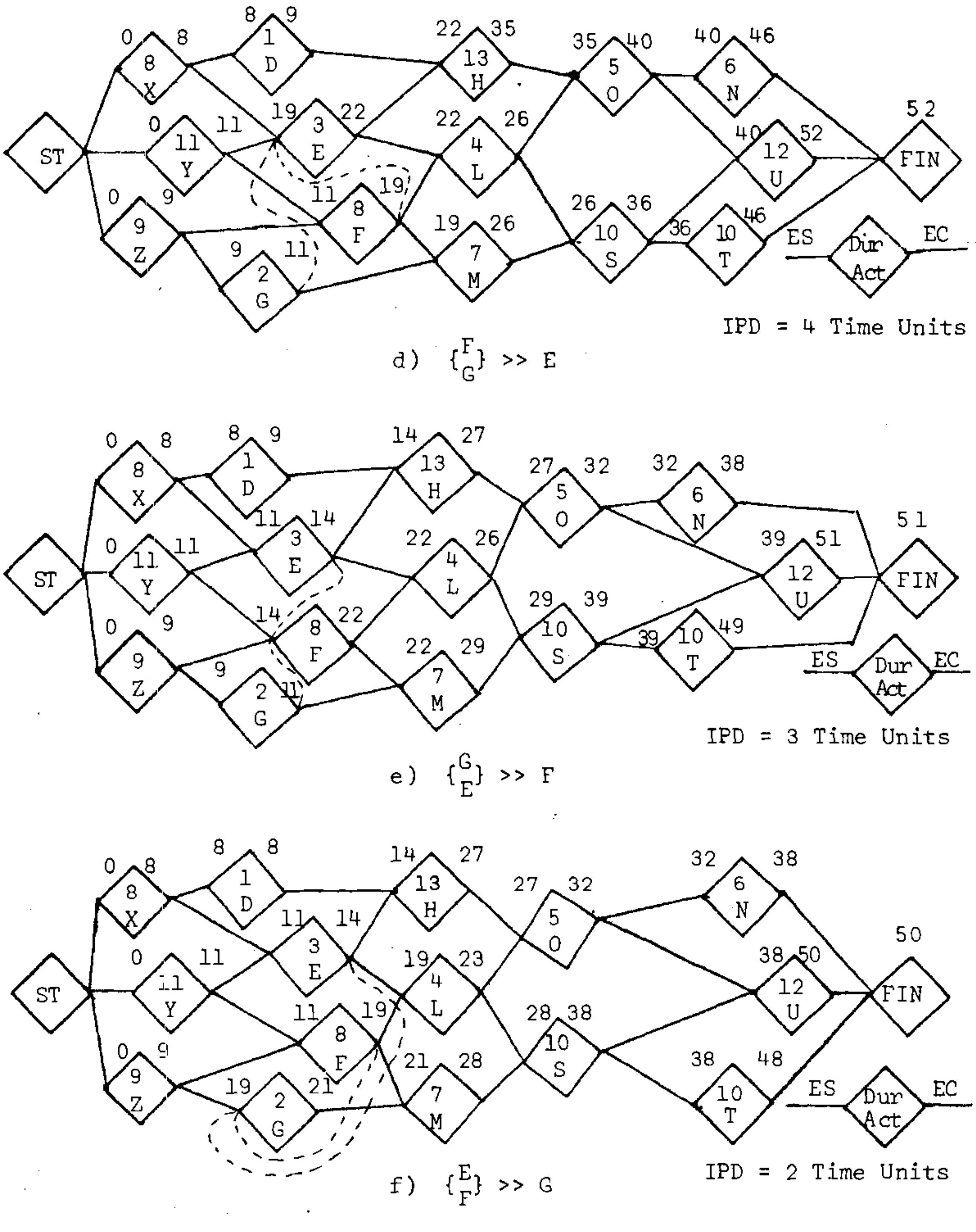
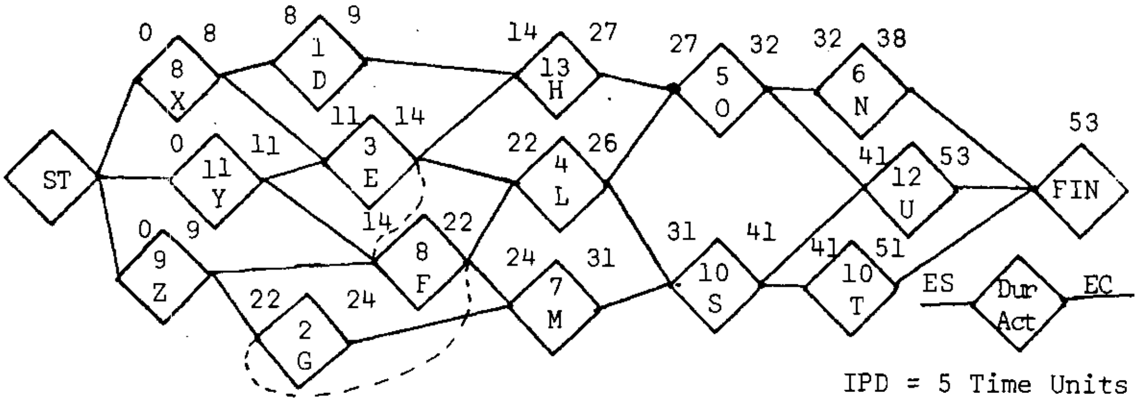
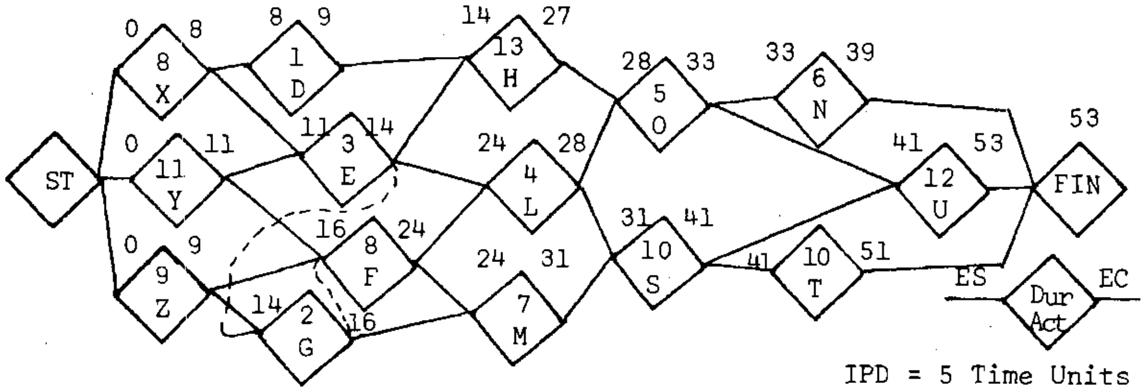


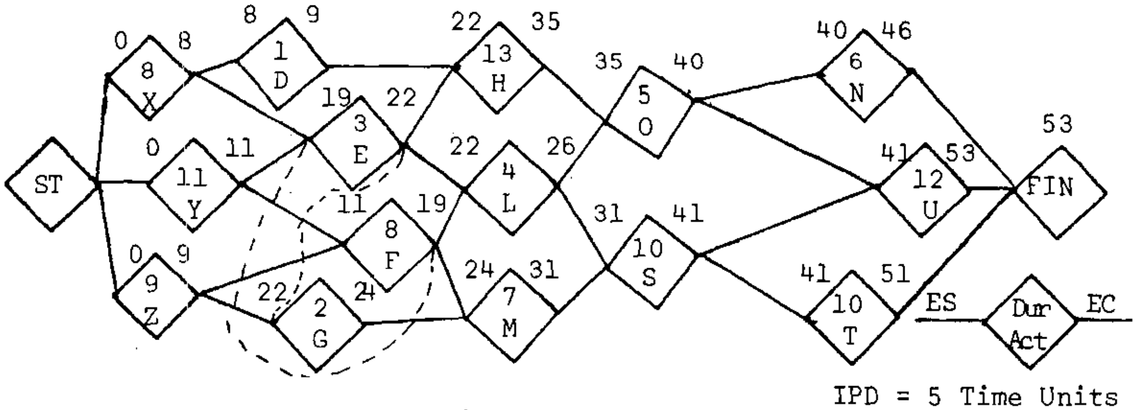
Figure 7. Separate Networks for Different Sequences Within the Nonsimultaneous Set (Continued)



g) E >> F >> G

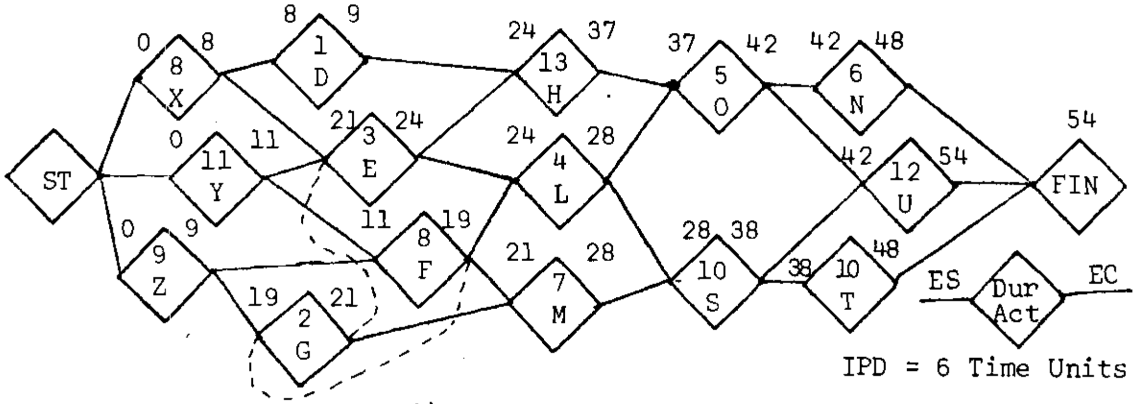


h) E >> G >> F

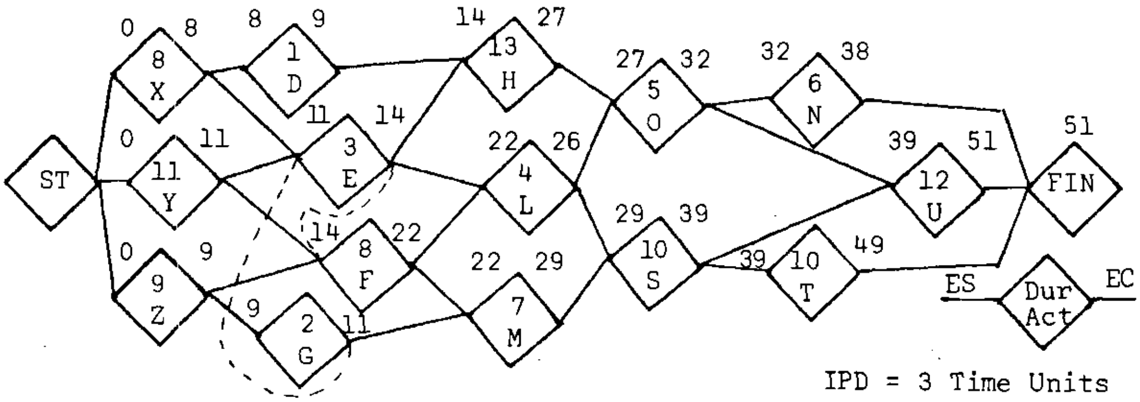


i) F >> E >> G

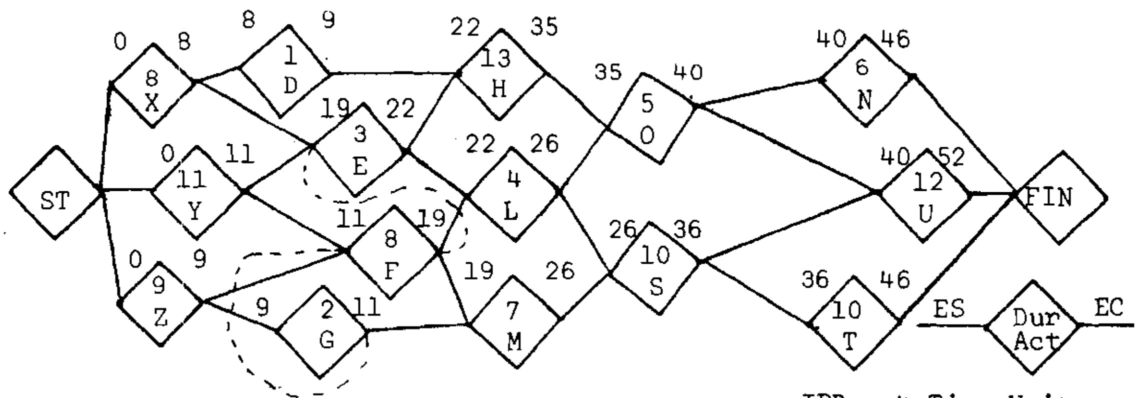
Figure 7. Separate Networks for Different Sequences Within the Nonsimultaneous Set (Continued)



j) F >> G >> E



k) G >> E >> F



l) G >> F >> E

Figure 7. Separate Networks for Different Sequences Within the Nonsimultaneous Set (Concluded)

the total project duration of the basic network which was constructed without considering the nonsimultaneity constraint, existing between the activities E, F and G as designated in the basic network.

Actual computation by constructing separate network for each feasible sequence of activities within the nonsimultaneous set, is now presented to indicate the desirability of following the worksheet presented in this chapter. Thus, for example, six separate networks are constructed and the forward pass made for each of these networks (Figure 7a through Figure 7f). For example two, six separate networks are again constructed and the forward pass made for each of these networks (Figures 7g through 7l).

Proof of Optimality of the Proposed Worksheet Approach

Suppose the three activities of the single nonsimultaneous set are I, J, K.

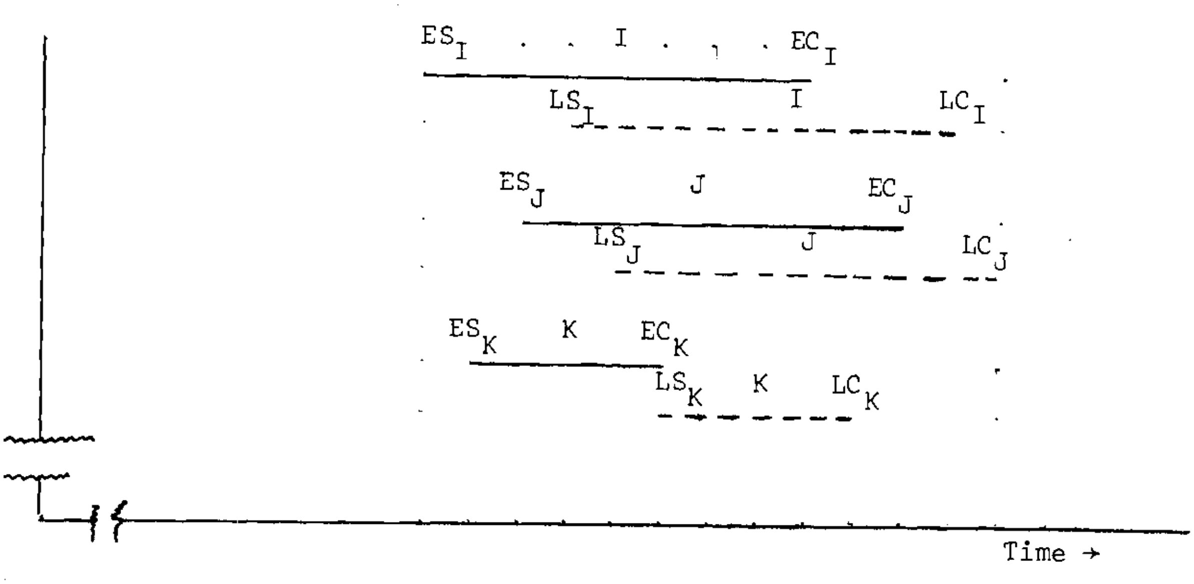


Figure 8. Generalized Representation of Three Activities in Their ES and LS Schedules

Let the solid lines indicate the earliest start time schedules of these three activities. This means that the activities are scheduled to start at their ES times and to complete at their EC times. Let the dotted lines represent the latest start schedules. That is, the activities, if scheduled to commence at their LS times and end at their LC times, then the dotted lines will indicate the time span when these take place.

Now, these ES, EC, LS and LC time values are based on the basic network which has all technological precedence relationships incorporated into it. Thus, it is implicit that we cannot have any activity start before its earliest start time, since the most critical of its predecessors will be complete only by the ES time of the activity in question. However, it is possible to postpone an activity beyond its latest start time though such a postponement will definitely extend the length of the project. And the increase in project duration due to a postponement of an activity beyond its LS time is equal to the time period through which such a postponement is made. This follows from the definition of LS.

Consider Figure 8 and the three activities I, J and K. Suppose we consider the sequence IJK (meaning I >> J >> K), to resolve the non-simultaneity between I, J and K. Then *I* can be scheduled at its ES time. Activity J cannot be started until activity I is completed due to the added precedence I >> J >> K. I is completed at EC_I. At this point of time, J can be started, if all its predecessors are completed. So the earliest possible start of activity J so as to satisfy I >> J >> K will be EC_I or ES_J, whichever is greater. If the latter is greater, then there is no chance of postponing activity J beyond its allowable LS. If, however, EC_I is greater than ES_J, there may be an increase in project duration by a quantity equal to $EC_I - LS_J$. But if EC_I is less than or equal to LS_J, this indicates that J can be started at EC_I without affecting the project length. So, the positioning of J to succeed I will increase the project length by a quantity

$$\text{MAX}\begin{bmatrix} 0 \\ EC_I - LS_J \end{bmatrix} = \alpha \text{ say.} \tag{18}$$

Now I and J are positioned. I is in progress from ES_I to EC_I. J is in progress from EC_I to $EC_I + d_J$, if EC_I is greater than ES_J; or from ES_J to $ES_J + d_J$ if ES_J is greater than EC_I, and the IPD due to this positioning is α. Now activity K cannot be started until J is completed and all the technological predecessors of K are completed. That is, K can be started only at time $\text{MAX}\binom{EC_I}{ES_J} + d_J$ or ES_K, whichever is greater. If this is less than LS_K, there is no effect on IPD, due to this positioning of K. So the IPD will be just equal to $\text{MAX}\binom{0}{EC_I - LS_J}$ which is due to the positioning of J to succeed I. However, if the quantity $\text{MAX}\binom{EC_I}{ES_J} + d_J$ is greater than LS_K, then this means that there will be an increase in project duration by a quantity equal to

$$\text{MAX}\binom{EC_I}{ES_J} + d_J - LS_K. \tag{19}$$

Thus, the positioning of K will result in an IPD of

$$\text{MAX}\left[\begin{array}{c} 0 \\ \text{MAX}\begin{pmatrix} EC_I \\ ES_J \end{pmatrix} + d_J - LS_K \end{array}\right. = \beta \text{ say.} \tag{20}$$

From the above discussions, two aspects are clear: 1) Positioning of J to succeed I will cause an IPD of $\text{MAX}\left\{\begin{array}{c} 0 \\ EC_I - LS_J \end{array}\right.$ where $EC_I - LS_J$ is a result of the postponement of activity J beyond its allowable latest start time; 2) Positioning of K to have precedence relationship of I >> J >> K will result in an IPD of

$$\text{MAX}\left[\begin{array}{c} 0 \\ \text{MAX}\begin{pmatrix} EC_I \\ ES_J \end{pmatrix} + d_J - LS_K \end{array}\right.$$

where $\text{MAX}\begin{pmatrix} EC_I \\ ES_J \end{pmatrix} + d_J - LS_K$ is the IPD as a result of the postponement of activity K beyond its allowable LS time. So, we have a situation where positioning of an activity J results in an IPD of α and positioning of another activity K in the same network results in an IPD of β which is independent of the value α. Hence, the project will be extended beyond its previous completion date, as calculated from the basic network, by an amount α or β, whichever is greater. Or, IPD due to resequencing within the single nonsimultaneous set equal to $\text{MAX}\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ or

$$= \text{MAX}\begin{bmatrix} \text{MAX}\begin{bmatrix} 0 \\ EC_I - LS_J \end{bmatrix} \\ \text{MAX}\begin{bmatrix} 0 \\ \text{MAX}\left(\begin{matrix} EC \\ ES_J \end{matrix} I\right) + d_J - LS_K \end{bmatrix} \end{bmatrix} \tag{21}$$

This is the quantity arrived at in line ten of the worksheet No. 1.

Thus, we have seen that for a single nonsimultaneous set of three activities, having a simultaneity maximum of one, the IPD is given by expression 21.

Consider the sequence of the form $X >> \{{Y \atop Z}\}$; that is, the situation when X precedes the set {Y,Z} and the activities Y and Z can be in progress simultaneously. Evidently, X will be scheduled at its ES and will be in progress during the time period ES_X through EC_X. At this point of time Y and/or Z can be started, if no technological precedence requirements are violated. Now, if both LS_Y and LS_Z are greater then EC_X, then there will be no increase in project length due to such a positioning, since, in this situation, commencement of neither Y nor Z is postponed beyond its latest start time. However, if EC_X is greater than either LS_Y or LS_Z, an increase in project length will result; and the increase will be equal to $EC_X - LS_Y$ or $EC_X - LS_Z$, whichever is greater. That is, the IPD due to the imposed precedence restraint $X >> \{{Y \atop Z}\}$, will be given by

$$\text{MAX}\begin{bmatrix} 0 \\ \text{MAX}\begin{bmatrix} EC_X - LS_Y \\ EC_X - LS_Z \end{bmatrix} \end{bmatrix}$$

or

$$\text{MAX}\begin{bmatrix} 0 \\ EC_X - \text{MIN}\binom{LS_Y}{LS_Z} \end{bmatrix} \tag{22}$$

For sequences of the form $X >> \{{Y \atop Z}\}$, namely, for $A >> \{{B \atop C}\}$, $B >> \{{C \atop A}\}$, $C >> \{{A \atop B}\}$, line ten in worksheet gives the value of expression 22.

Now, for sequences of form $\{{P \atop Q}\} >> R$, an expression for IPD will be derived. With this imposed precedence R can start only after both P and Q are completed. Both P and Q are at their earliest start schedules. Then R cannot be started until time EC_P or EC_Q, whichever is the greater. And if the $\text{MAX}\binom{EC_P}{EC_Q}$ is greater than LS_R, this causes postponement of activity R beyond its LS by an amount $\text{MAX}\binom{EC_P}{EC_Q} - LS_R$. However, if $\text{MAX}\binom{EC_P}{EC_Q}$ is less than LS_R, R is not postponed beyond its latest start time. Thus, IPD for sequences of form $\{{P \atop Q}\} >> R$, will be given by the expression

$$\text{MAX}\begin{bmatrix} 0 \\ \text{MAX}\binom{EC_P}{EC_Q} - LS_R \end{bmatrix} \tag{23}$$

Modified Version of the Proposed Approach for Single Nonsimultaneous Sets of Simultaneity Maximum Equal to One

As we have seen, although the proposed approach using the worksheet has simplified the computations to a considerable extent, we are required to calculate IPD for *all* feasible combinations as determined by the occurrence of Case I, II, III, or IV, in order to find the optimal sequence. The worksheet approach is useful for the project manager to get an idea as to how the project length will be affected as a result of alternative sequences within the single nonsimultaneous set. This will enable him to objectively evaluate the effect of the nonsimultaneity on the over-all project length. However, if it is just desired to arrive at *an* optimal solution without necessarily knowing relative merits and demerits of alternative sequences, from the project length considerations, then, any procedure to achieve this, will be of immense help to the project manager. This procedure should be such that it eliminates the need of calculating IPD's for all feasible combinations, thereby simplifying further the approach using worksheet.

Modified Approach

The forward pass and the backward pass on the basic network are done. It may be recalled that the basic network is the one that is constructed to depict all the true or technological precedence requirements, but not the nonsimultaneity restraint. Now all the three members of the single nonsimultaneous set are identified. Suppose A, B, C are these three members. Our objective is to find a sequence within the nonsimultaneous set which will tend to result in *an* over-all *optimal* schedule. This procedure is heuristic in nature and has not been proved to always lead to an optimum.

Information Required for Implementation

a) Basic network incorporating all technological precedence relationships and the ES, EC, LS, LC, d values of all the activities obtained after the passes on this basic network.

b) Information regarding resource requirements and availabilities, of the three activities in the single nonsimultaneous set.

Having identified the three members of the nonsimultaneous set and knowing their ES, EC, LS and LC times, comparison is made between the *boundary sum* values of the members of the nonsimultaneous set. Boundary sum (BS) for an activity is defined as the sum of its ES and LC times. Thus,

$$BS = ES + LC = LS + EC \qquad (24)$$

The selection of *boundary sum* as a criterion to choose the sequence, was intuitive. This was based on an apprehension that the boundary sum, being the sum of ES and LC of an activity reflects to an extent: a) the domain in which the activity can be scheduled and also b) the location of this domain with respect to the other activities. Suppose the *boundary sum* values of activities A, B, C are denoted by BS_A, BS_B, and BS_C, respectively. To find an optimal solution, the following steps are followed as guidelines.

Step 1: If $BS_A \neq BS_B \neq BS_C$, then do the following. Otherwise, go to step 2.

a) Choose the sequence such that the activity having MIN BS associated with it is scheduled to occur first within the nonsimultaneous set. The activity with next higher BS value is scheduled to immediately succeed this. And the activity with maximum BS is positioned at the end. Thus, if $BS_A < BS_B < BS_C$, then the sequence ABC is considered as the optimal one.

Step 2: If two of the three BS values are equal and this is less than the BS value of the third, then do the following. Otherwise, go to step 3.

a) Position the activity having greater BS, toward the end in the sequence. The two others have not been positioned.

b) Add the pseudo-precedence requirements in the basic network such that the positioned activity mentioned in a) above is an immediate successor to both of the unpositioned activities in the nonsimultaneous set.

c) Call the network *modified network* and make a forward pass and a backward pass on this modified network.

d) Compute the two new BS values for the two activities which have not been positioned yet. Call them BS'_x where x corresponds to either of the unpositioned activities.

e) Now position these two activities such that BS' of the successor is greater than or at least equal to BS' of the predecessor. Thus, a sequence which is *good* is obtained.

Step 3: If two of the three BS values are equal and this is greater than the BS value of the third, then do the following. Otherwise, go to step 4.

a) Position the activity having MIN BS value at the beginning of the sequence. The two other activities have not been positioned.

b) Add the pseudo-precedence requirement in the basic network such that the positioned activity mentioned in (a) above is an immediate predecessor to both of the unpositioned activities in the nonsimultaneous set.

c) Call this network *modified network* and make a forward pass and a backward pass on the modified network.

d) Compute the two new BS values for the two activities which have not been positioned yet. Call them BS'_x where x corresponds to either of the unpositioned activities.

e) Now position these two activities in an order such that BS' of the successor is greater than or at least equal to BS' of the predecessor.

Thus, the three activities in the nonsimultaneous set are positioned.

Step 4: If all the three BS values are equal, then do the following.

a) Position one of the activities at the end of the sequence and add the pseudo-precedence in the basic network such that the positioned activity is an immediate successor to the other two.

b) Do substeps c, d and e of step 2.

Thus, the three activities in the nonsimultaneous set are positioned.

Example Application

Example One

Consider the network in Figure 6, page 58. Suppose E, F, G are the members of the single nonsimultaneous set, of simultaneity maximum equal to one. From the basic network calculations shown in the figure, we have

$$\begin{aligned} \text{for activity E, } BS_E &= ES_E + LC_E = 11 + 18 = 29 \\ \text{F, } BS_F &= ES_F + LC_F = 11 + 19 = 30 \\ \text{G, } BS_G &= ES_G + LC_G = 9 + 19 = 28 \end{aligned}$$

To find a *good* sequence:

Step 1: $(BS_E = 29) \neq (BS_F = 30) \neq (BS_G = 28)$. So do the following.

a) Choose the sequence such that the activity having the MIN BS value, that is, activity G with $BS_G = 28$, is scheduled first. Then the activity E which has the next higher BS ($BS_E = 29$) is scheduled to immediately succeed G. And the activity with maximum BS, that is F with BS = 30 is positioned at the end. Then, the sequence is GEF. It may be noted that this checks with the solution by the worksheet approach.

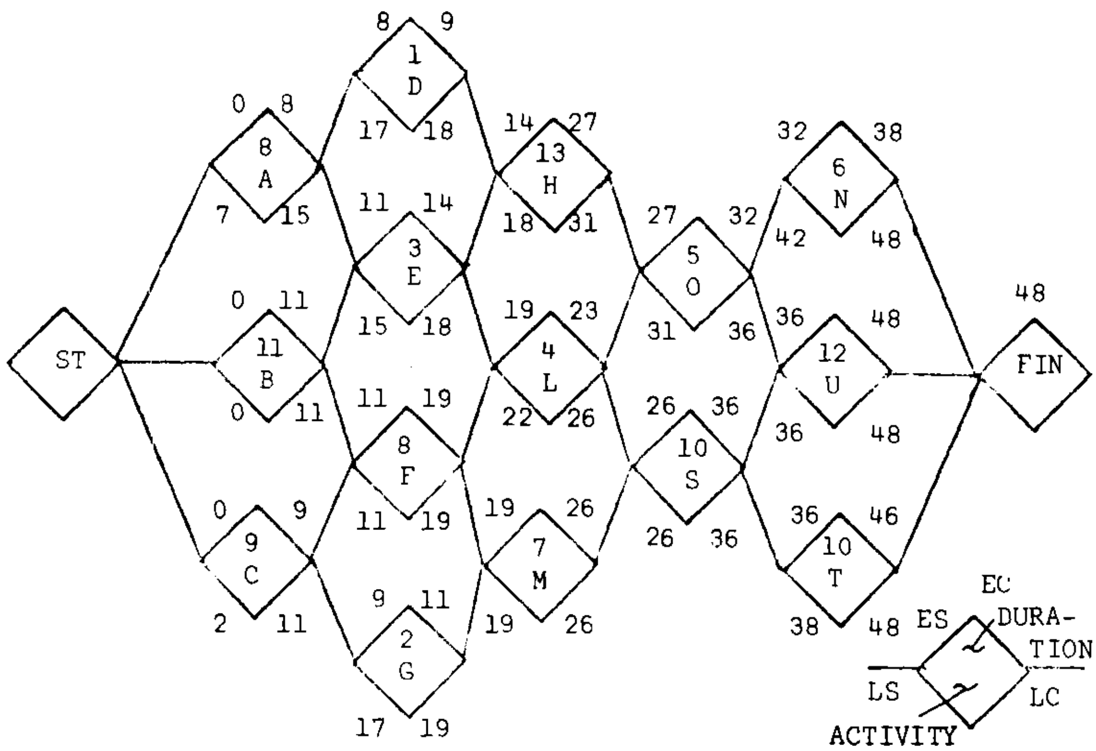


Figure 9. Basic Network for Example 2, Page 83

Example Two

Consider the network in Figure 9. Assume that A, B, C of this network form the single nonsimultaneous set with a simultaneity maximum of one. From the calculations on the basic network, we have

$$ES_A = 0 \qquad LC_A = 15$$

$$ES_B = 0 \qquad LC_B = 11$$

$$ES_C = 0 \qquad LC_C = 11$$

From these, BS values can be calculated. $BS_A = 15$; $BS_B = 11$; $BS_C = 11$. To find a *good* solution:

Step 1: Not applicable since $BS_B = BS_C$. So go to step 2.

Step 2: $BS_B = BS_C = 11$ and is less than $BS_A = 15$. So do the following.

a) Position A, which has greater BS, toward the end. B and C have not been positioned.

b) Modify the network by adding the pseudo-precedence requirement $\{^B_C\} >> A$. The following network results

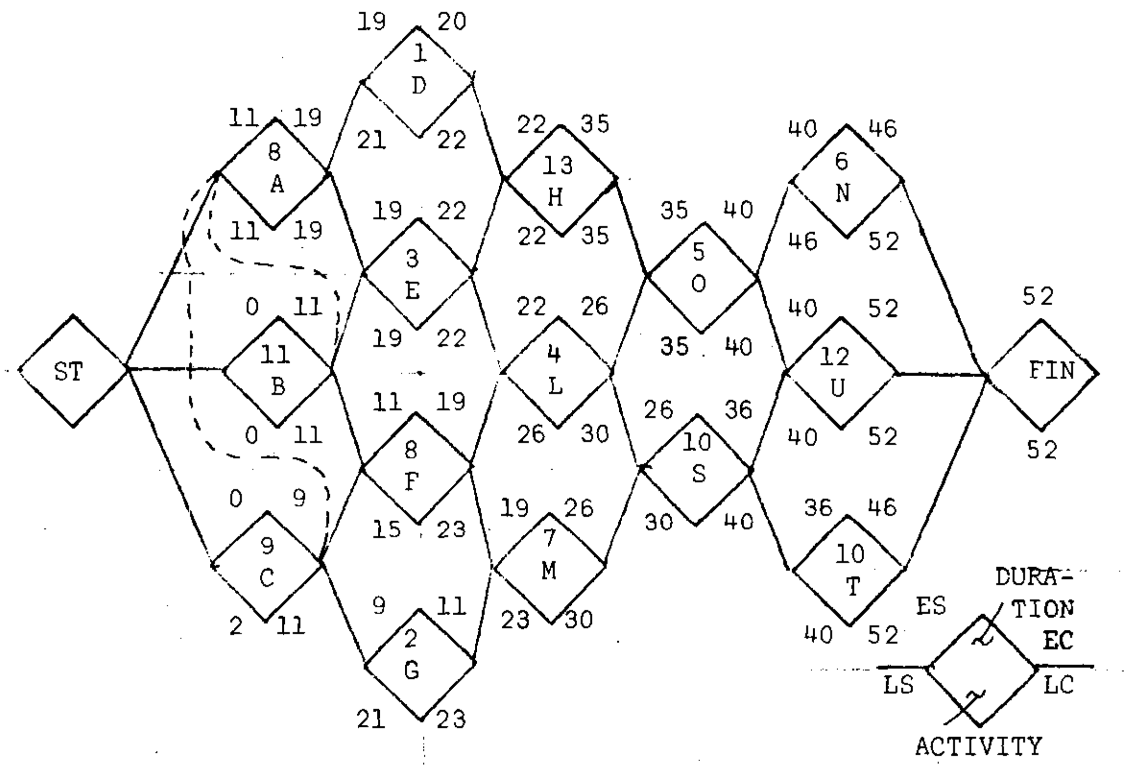


Figure 10. Modified Network $\{^B_C\} >> A$

c) Make a forward and backward pass on this modified network. Obtain the values of $ES'_B = 0$, $LC'_B = 11$; $ES'_C = 0$, $LC'_C = 11$.

d) Compute the values of BS'_B and BS'_C. $BS'_B = 11$; $BS'_C = 11$.

e) Now since these two BS' values are equal, we can have either B succeed C or C to succeed B. Thus, we have BCA and CBA as *good* sequences.

Example (3)

Consider the following network

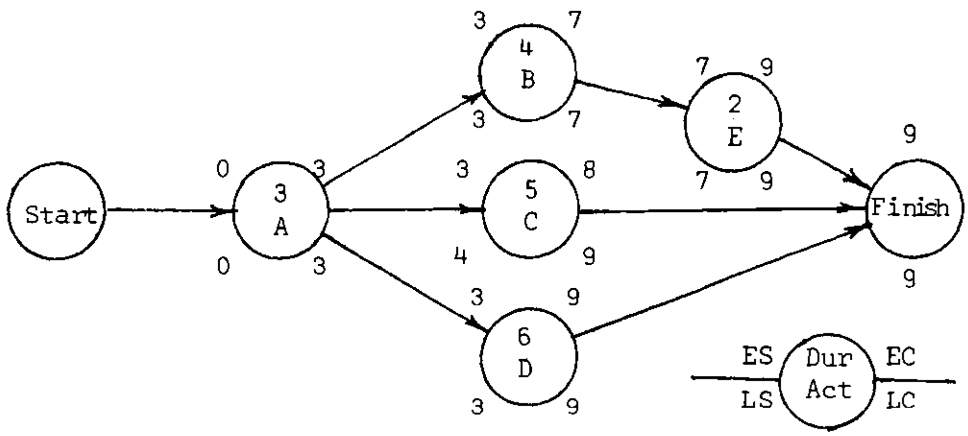


Figure 11. Basic Network for Example 3

Suppose B, C, D are the members of the single nonsimultaneous set. From the calculations on the basic network, we have $BS_B = 10$, $BS_C = 12$, $BS_D = 12$. To find a *good* sequence:

Step 1: Not applicable since $BS_C = BS_D = 12$. Go to step 2.

Step 2: $BS_C = BS_D = 12$, is greater than $BS_B = 10$. So go to step 3.

Step 3: $BS_C = BS_D = 12$ and is greater than $BS_B = 10$, so do the following:

a) Position the activity B having MIN BS value at the beginning of the sequence.

b) Construct a modified network with pseudo-precedence $B >> \{{C \atop D}\}$.

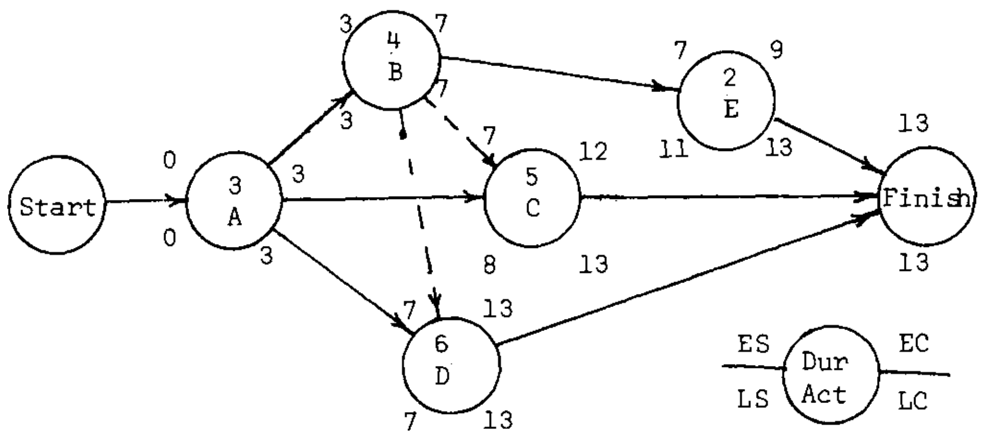


Figure 12. Modified Network with $B >> \{{C \atop D}\}$

c) Make a forward pass and a backward pass on this modified network. Obtain the ES' and LC' values for C and D.

$ES'_C = 7 \qquad LC'_C = 13$

$ES'_D = 7 \qquad LC'_D = 13$

d) $BS'_C = 20 \qquad BS'_D = 20$

e) Since the two BS' values are equal, we can have either C to succeed D or D to succeed C. Thus BCD and BDC are found *good*. This can be verified by the worksheet approach.

Example (4)

Consider the following network.

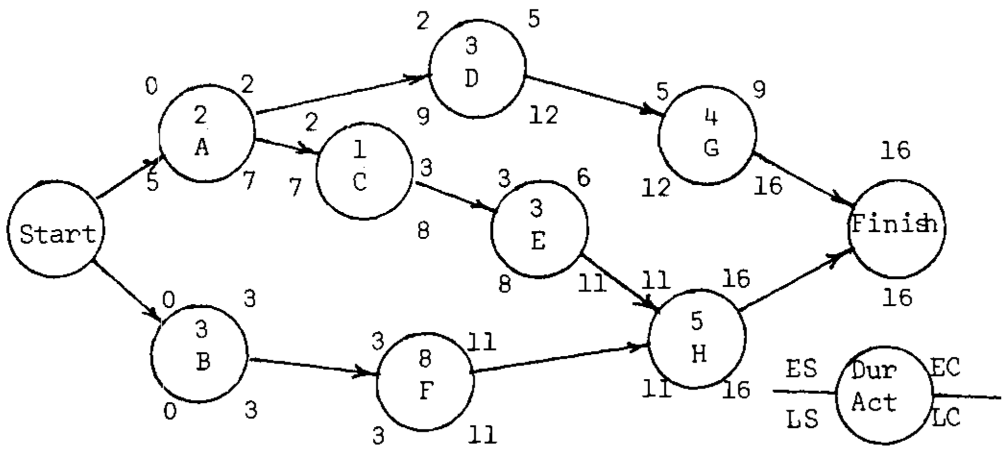


Figure 13. Basic Network for Example 4

Suppose D, E, F are the members of the single nonsimultaneous set. From the basic network calculations, we have $BS_D = 14$; $BS_E = 14$; $BS_F = 14$.

To find a *good* sequence:

Steps 1, 2 and 3: Not applicable since $BS_D = BS_E = BS_F$. Go to step 4.

Step 4: Since $BS_D = BS_E = BS_F$, do the following.

a) Position one of the activities at the end of the sequence. Let F be scheduled at the end. Construct the modified network with $\{{}^{D}_{E}\} >> F$. The following network results.

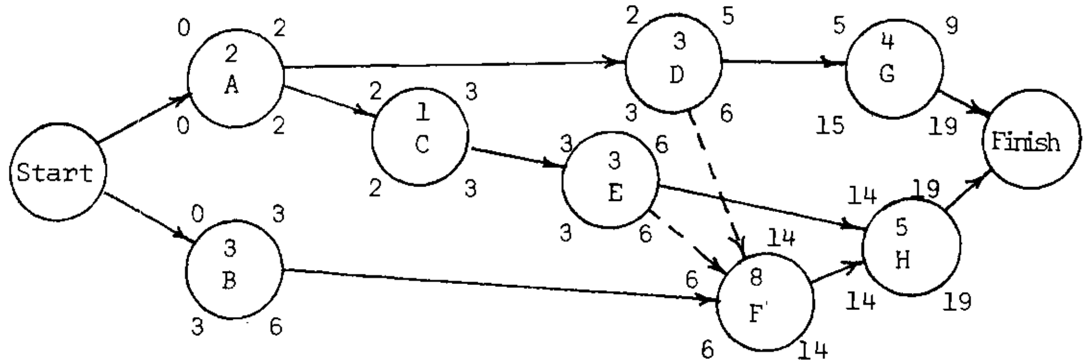


Figure 14. Modified Network with $\{{}^{D}_{E}\} >> F$

b) Go to substeps c, d, e of step 2. That is,

c) Make a forward pass and backward pass on the modified network. Get the ES' and LC' values for D and E

$$ES'_D = 2 \qquad LC'_D = 6$$

$$ES'_E = 3 \qquad LC'_E = 6$$

d) $BS'_D = 8 \qquad BS'_E = 9$

e) Position D and E such that D precedes E since $BS'_D < BS'_E$. So DEF is a *good* sequence. This can be verified by the worksheet.

Computational Magnitude of the Proposed Approaches as Compared with Complete Enumeration of all Possible Networks

As the number of activities in the basic network increases, the usefulness of the proposed approaches is more striking. For the complete enumeration approach, if there are T activities in the basic network, we have to construct 12 networks each with T activities. Thus, we have to deal with 12 T activities. With the proposed worksheet approach, we deal only with T activities in the basic network plus three activities in the worksheet. The relatively simple computation in the worksheet leads to selection of an optimal sequence.

With the modified approach using BS values, the computation is even much more simple. At worst, we have to construct only two networks including the basic network and the forward and the backward pass on these two networks will indicate an optimal sequence.

CHAPTER IV

SINGLE NONSIMULTANEOUS SET OF FOUR ACTIVITIES WITH A SIMULTANEITY MAXIMUM EQUAL TO ONE

Introduction

In this chapter, a procedure is given to optimally resolve the nonsimultaneity existing between four activities in a single nonsimultaneous set. The concepts used are basically quite similar to those in Chapter III. However, only the situations involving a single nonsimultaneous set of four activities with a simultaneity maximum equal to one are considered. A worksheet approach is presented which is used to examine all feasible combinations and then select the sequence or sequences which result in minimum over-all project length. Four activities in the nonsimultaneous set can be sequenced in 4! = 24 different ways. The IPD due to each of these different sequences is computed using the worksheet and the sequence which results in minimum IPD is chosen as the optimal. A modified version of this procedure is presented and this modified version makes use of the *boundary sum* values. This version simplifies the computations considerably. Though it may not always lead to an optimal solution, it does tend to indicate a near-optimal solution, at the worst.

Proposed Worksheet Approach

From the calculations on the basic network, the ES, EC, LS and LC values of all the activities are known. The project length disregarding the nonsimultaneity constraint is also known. Then the minimum IPD due to resolution of the nonsimultaneity constraint is determined by computing IPD for each feasible sequencing combination within the nonsimultaneous set. To implement this procedure, worksheet No. 2 is used in conjunction with the following guidelines.

Step 1: Identify the four activities in the single nonsimultaneous set. Enter the ES, EC, LS, LC, d and BS values of these four activities in the table on the top of the worksheet.

Step 2: Feasible sequences are indicated on the top of each column. The sequences are of the general form IJKL which indicates precedence I >> J >> K >> L. Enter the EC_I values in line one of worksheet. Enter the LS_J values in line two.

Step 3: For each column, subtract the quantity in line two from that in line one and if result is nonnegative, enter it in line three. Otherwise, enter zero.

Step 4: Enter ES_J values in line four. For each column, compare the quantities in line one and line four, and enter the greater of the two in line five.

Step 5: Enter d_J values in line six and LS_K values in line seven.

Table 5. Blank Worksheet No. 2, Sheet 1

	EC	LS	ES	d	BS	Activity identifier
A						
B						
C						
D						

LINE		ABCD	ABDC	ACBD	ACDB	ADBC	ADCB	BACD	BADC	BCAD	BCDA	BDAC	BDCA
1	EC	A	A	A	A	A	A	B	B	B	B	B	B
2	LS	B	B	C	C	D	D	A	A	C	C	D	D
3	MAX $\{ {0 \atop 1-2}$												
4	ES	B	B	C	C	D	D	A	A	C	C	D	D
5	MAX $\{ {1 \atop 4}$												
6	d	B	B	C	C	D	D	A	A	C	C	D	D
7	LS	C	D	B	D	B	C	C	D	A	D	A	C
8	5 + 6 - 7												
9	MAX $\{ {0 \atop 8}$												
10	1 + 6												
11	EC	B	B	C	C	D	D	A	A	C	C	D	D
12	ES	C	D	B	D	B	C	C	D	A	D	A	C
13	MAX $\{ \begin{smallmatrix} 10 \\ 11 \\ 12 \end{smallmatrix}$												
14	d	C	D	B	D	B	C	C	D	A	D	A	C
15	LS	D	C	D	B	C	B	D	C	D	A	C	A
16	13 + 14 - 15												
17	MAX $\{ {0 \atop 16}$												
18	MAX $\{ \begin{smallmatrix} 3 \\ 9 \\ 17 \end{smallmatrix}$												

Table 6. Blank Worksheet No. 2, Sheet 2

	EC	LS	ES	d	BS	Activity identifier
A						
B						
C						
D						

LINE		CABD	CADB	CBAD	CBDA	CDAB	CDBA	DABC	DACB	DBAC	DBCA	DCAB	DCBA
1	EC	C	C	C	C	C	C	D	D	D	D	D	D
2	LS	A	A	B	B	D	D	A	A	B	B	C	C
3	MAX $\{ \begin{matrix} 0 \\ 1 - 2 \end{matrix}$												
4	ES	A	A	B	B	D	D	A	A	B	B	C	C
5	MAX $\{ \begin{matrix} 1 \\ 4 \end{matrix}$												
6	d	A	A	B	B	D	D	A	A	B	B	C	C
7	LS	B	D	A	D	A	B	B	C	A	C	A	B
8	5 + 6 - 7												
9	MAX $\{ \begin{matrix} 0 \\ 8 \end{matrix}$												
10	1 + 6												
11	EC	A	A	B	B	D	D	A	A	B	B	C	C
12	ES	B	D	A	D	A	B	B	C	A	C	A	B
13	MAX $\{ \begin{matrix} 10 \\ 11 \\ 12 \end{matrix}$												
14	d	B	D	A	D	A	B	B	C	A	C	A	B
15	LS	D	B	D	A	B	A	C	B	C	A	B	A
16	13 + 14 - 15												
17	MAX $\{ \begin{matrix} 0 \\ 16 \end{matrix}$												
18	MAX $\{ \begin{matrix} 3 \\ 9 \\ 17 \end{matrix}$												

Step 6: For each column, add the quantities in line five and line six and from this sum, subtract the quantity in line seven. Enter the result in line eight.

Step 7: For each column, if the quantity in line eight is nonnegative, enter the same in line nine. Otherwise, enter zero in line 9.

Step 8: For each column, add the quantities in line one and line six, and enter the sum in line ten.

Step 9: Enter the values of EC_J in line 11 and ES_K in line 12.

Step 10: For each column, add the quantities in lines 10, 11, 12 and enter the maximum of these in line 13.

Step 11: Enter d_K and LS_L values in line 14 and 15, respectively.

Step 12: For each column, add the quantities in line 13, 14 and subtract from this sum, the quantity in line 15. Enter the result in line 16.

Step 13: For each column, if the quantity in line 16 is nonnegative, enter the same in line 17. Otherwise, enter zero in line 17.

Step 14: For each column, compare the quantities in lines 3, 9, 17 and enter the maximum in line 18. Line 18 gives the IPD for each sequence.

Step 15: Choose the sequence which results in minimum IPD. This is optimal. If there is more than one sequence with the same minimum IPD, those sequences are all optimal.

Example Application

Consider the network shown below.

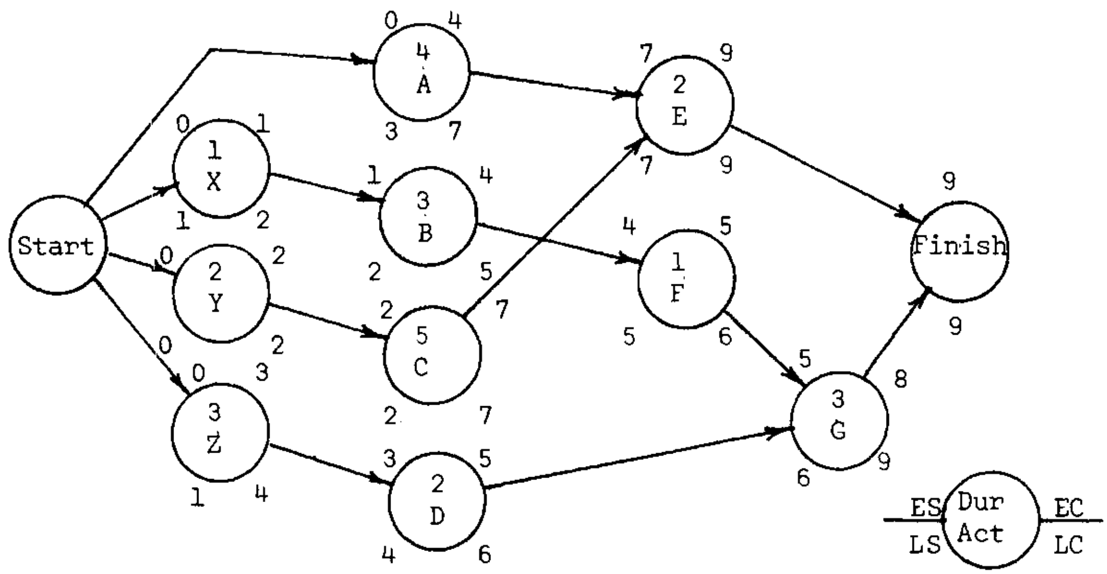


Figure 15. Basic Network with a Four-Activity Nonsimultaneous Set--Example Application Using Worksheet Approach

From the calculations on the basic network, the ES, EC, LS and LC values of all the activities are known. To find the optimal sequence, use the worksheet No. 2, compute IPD for each sequence and then choose the sequence with minimum IPD. Detailed computations for finding IPD for sequence ABCD are presented.

Step 1: Identify the four activities in the single nonsimultaneous set as A, B, C and D in the network. Enter the ES, EC, LS, LC, d and BS values of these four activities in the table on the top of the worksheet.

Table 7. Worksheet No. 2, Sheet 1, Used for Sequencing the Four Activities of a Single Nonsimultaneous Set

	EC	LS	ES	d	BS	Activity identifier
A	4	3	0	4	7	A
B	4	2	1	3	6	B
C	7	2	2	5	9	C
D	5	4	3	2	9	D

LINE		ABCD	ABDC	ACBD	ACDB	ADBC	ADCB	BACD	BADC	BCAD	BCDA	BDAC	BDCA
1	EC	A 4	A 4	A 4	A 4	A 4	A 4	B 4	B 4	B 4	B 4	B 4	B 4
2	LS	B 2	B 2	C 2	C 2	D 4	D 4	A 3	A 3	C 2	C 2	D 4	D 4
3	$\text{MAX}\left\{\begin{matrix}0\\1-2\end{matrix}\right.$	2	2	2	2	0	0	1	1	2	2	0	0
4	ES	B 1	B 1	C 2	C 2	D 3	D 3	A 0	A 0	C 2	C 2	D 3	D 3
5	$\text{MAX}\left\{\begin{matrix}1\\4\end{matrix}\right.$	4	4	4	4	4	4	4	4	4	4	4	4
6	d	B 3	B 3	C 5	C 5	D 2	D 2	A 4	A 4	C 5	C 5	D 2	D 2
7	LS	C 2	D 4	B 2	D 4	B 2	C 2	C 2	D 4	A 3	D 4	A 3	C 2
8	5 + 6 - 7	5	3	7	5	4	4	6	4	6	5	3	4
9	$\text{MAX}\left\{\begin{matrix}0\\8\end{matrix}\right.$	5	3	7	5	4	4	6	4	6	5	3	4
10	1 + 6	7	7	9	9	6	6	8	8	9	9	6	6
11	EC	B 4	B 4	C 7	C 7	D 5	D 5	A 4	A 4	C 7	C 7	D 5	D 5
12	ES	C 2	D 3	B 1	D 3	B 1	C 2	C 2	D 3	A 0	D 3	A 0	C 2
13	$\text{MAX}\left\{\begin{matrix}10\\11\\12\end{matrix}\right.$	7	7	9	9	6	6	8	8	9	9	6	6
14	d	C 5	D 2	B 3	D 2	B 3	C 5	C 5	D 2	A 4	D 2	A 4	C 5
15	LS	D 4	C 2	D 4	B 2	C 2	B 2	D 4	C 2	D 4	A 3	C 2	A 3
16	13 + 14 - 15	8	7	8	9	7	9	9	8	9	8	8	8
17	$\text{MAX}\left\{\begin{matrix}0\\16\end{matrix}\right.$	8	7	8	9	7	9	9	8	9	8	8	8
18	$\text{MAX}\left\{\begin{matrix}3\\9\\17\end{matrix}\right.$	8	7	8	9	7	9	9	8	9	8	8	8

Table 8. Worksheet No. 2, Sheet 2, Used for Sequencing the Four Activities of a Single Nonsimultaneous Set

	EC	LS	ES	d	BS	Activity identifier
A	4	3	0	4	7	A
B	4	2	1	3	6	B
C	7	2	2	5	9	C
D	5	4	3	2	9	D

LINE		CABD	CADB	CBAD	CBDA	CDAB	CDBA	DABC	DACB	DBAC	DBCA	DCAB	DCBA
1	EC	C 7	C 7	C 7	C 7	C 7	C 7	D 5	D 5	D 5	D 5	D 5	D 5
2	LS	A 3	A 3	B 2	B 2	D 4	D 4	A 3	A 3	B 2	B 2	C 2	C 2
3	MAX { 0, 1 - 2	4	4	5	5	3	3	2	2	3	3	3	3
4	ES	A 0	A 0	B 1	B 1	D 3	D 3	A 0	A 0	B 1	B 1	C 2	C 2
5	MAX { 1, 4	7	7	7	7	7	7	5	5	5	5	5	5
6	d	A 4	A 4	B 3	B 3	D 2	D 2	A 4	A 4	B 3	B 3	C 5	C 5
7	LS	B 2	D 4	A 3	D 4	A 3	B 2	B 2	C 2	A 3	C 2	A 3	B 2
8	5 + 6 - 7	9	7	7	6	6	7	7	7	5	6	7	8
9	MAX { 0, 8	9	7	7	6	6	7	7	7	5	6	7	8
10	1 + 6	11	11	10	10	9	9	9	9	8	8	10	10
11	EC	A 4	A 4	B 4	B 4	D 5	D 5	A 4	A 4	B 4	B 4	C 7	C 7
12	ES	B 1	D 3	A 0	D 3	A 0	B 1	B 1	C 2	A 0	C 2	A 0	B 1
13	MAX { 10, 11, 12	11	11	10	10	9	9	9	9	8	8	10	10
14	d	B 3	D 2	A 4	D 2	A 4	B 3	B 3	C 5	A 4	C 5	A 4	B 3
15	LS	D 4	B 2	D 4	A 3	B 2	A 3	C 2	B 2	C 2	A 3	B 2	A 3
16	13 + 14 - 15	10	11	10	9	11	9	10	12	10	10	12	10
17	MAX { 0, 16	10	11	10	9	11	9	10	12	10	10	12	10
18	MAX { 3, 9, 17	10	11	10	9	11	9	10	12	10	10	12	10

Step 2: In the column for ABCD, enter $EC_A = 4$ in line one. Enter $LS_B = 2$ in line two.

Step 3: Subtract the quantity in line two from the quantity in line one. Thus, $4 - 2 = 2$, which is greater than 0. So enter 2 in line three.

Step 4: Enter $ES_B = 1$ in line four. Compare the quantities in line one and line four. That is, 4 and 1 out of which 4 is the greater. So enter 4 in line five.

Step 5: Enter $d_B = 3$ in line 6 and $LS_C = 2$ in line seven.

Step 6: Add the quantity in line five and that in line six and from this sum, subtract the quantity in line seven. That is, add 4 and 3 and from this sum of 7, subtract 2 and get the result as 5. So enter 5 in line eight.

Step 7: If the quantity in line eight is nonnegative, enter the same in line nine. Otherwise, enter zero in line nine. Five is nonnegative. So enter 5 in line nine.

Step 8: Add the quantity in line one and that in line six and enter the sum in line ten. Thus, add 4 and 3 and enter the sum 7 in line ten.

Step 9: Enter $EC_B = 4$ in line 11 and $ES_C = 2$ in line 12.

Step 10: Compare the quantities in lines 10, 11, 12 and enter the maximum of these in line 13. That is, compare 7, 4, 2 and enter 7 in line 13.

Step 11: Enter $d_C = 5$ in line 14 and $LS_D = 4$ in line 15.

Step 12: Add 7 in line 13 and 5 in line 14 and from this sum of 12, subtract 4 in line 15. Enter the result 8 in line 16.

Step 13: Since 8 in line 16 is nonnegative, enter the same 8 in line 17.

Step 14: Compare the values in lines 3, 9, 17; that is, 2, 5 and 8; and enter the maximum 8 in line 18. Line 18 gives the IPD. So IPD for sequence ABCD in the example is *8 time units.* Applying the same procedure to all other feasible combinations, the IPD for each can be computed. The following table shows the IPD for each combination.

ABCD = 8	BACD = 9	CABD = 10	DABC = 10
ABDC = 7	BADC = 8	CADB = 11	DACB = 12
ACBD = 8	BCAD = 9	CBAD = 10	DBAC = 10
ACDB = 9	BCDA = 8	CBDA = 9	DBCA = 10
ADBC = 7	BDAC = 8	CDAB = 11	DCAB = 12
ADCB = 9	BDCA = 8	CDBA = 9	DCBA = 10

Step 15: The sequences which have minimum IPD are ABDC and ADBC, both with an IPD of seven time units. So ABDC and ADBC are optimal sequences.

Proof of Optimality of the Foregoing Procedure

Let the solid lines indicate Earliest Start time schedules of the four activities of the single nonsimultaneous set of simultaneity maximum equal to one. Let the dotted lines represent the Latest Start schedule.

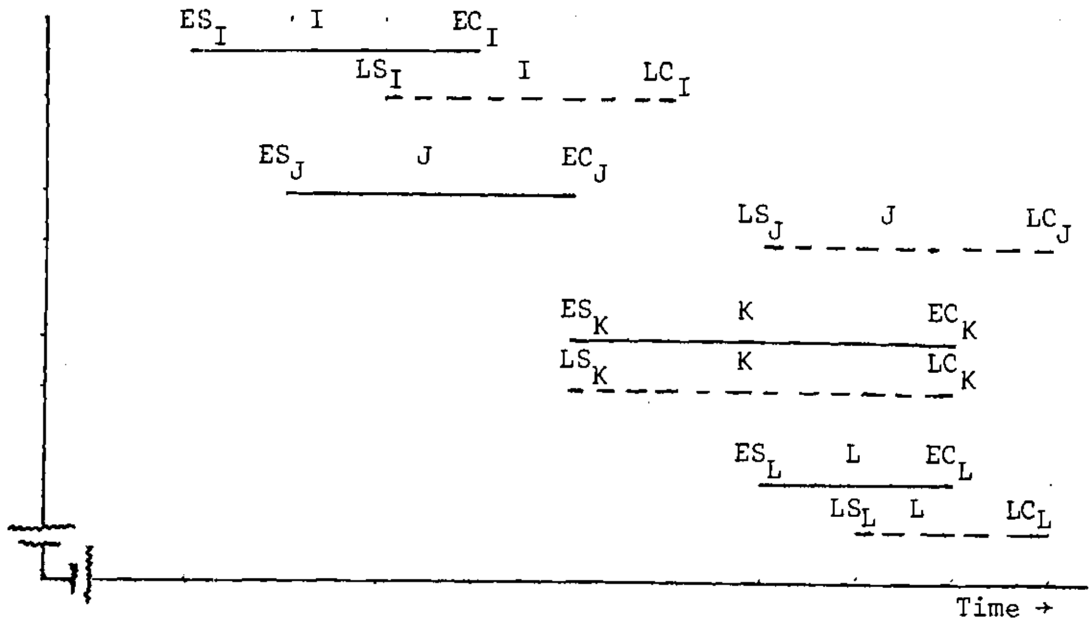


Figure 16. Generalized Representation of Four Activities in Their ES and LS Schedules

Now the ES, EC, LS and LC time values are based on the *basic network* which have all technological precedence relationships incorporated into it. To find an expression for IPD due to sequencing the four activities within the nonsimultaneous set, it is necessary to find out whether any of the four activities in question has been postponed beyond its LS and if so, by what amount.

Consider the sequence IJKL which has the imposed precedence $I \gg J \gg K \gg L$. In the last chapter, we have determined the IPD which is caused as a result of the positioning of J and K, which is given by the expression

$$\text{MAX}\begin{bmatrix} \text{MAX}\begin{bmatrix} 0 \\ (EC_I - LS_J) \end{bmatrix} \\ \text{MAX}\begin{bmatrix} 0 \\ \text{MAX}\begin{pmatrix} EC_I \\ ES_J \end{pmatrix} + d_J - LS_K \end{bmatrix} \end{bmatrix} \tag{25}$$

Now this expression holds good in the case of four activities also, if and only if activity L has not been forced to be postponed beyond its LS time. In case L has been postponed beyond its LS time, it is necessary to find out the period of time through which such a postponement has been made.

Depending on the interrelationships between the ES, EC, LS, LC and d values, the imposed precedence I >> J >> K >> L will alter the positioning of the four activities.

I is in progress from ES_I to EC_I.

At this point of time, J can be started if and only if EC_I is greater than or equal to ES_J. If not, starting of J has to be delayed till ES_J. So J will be started at time $\text{MAX}(^{EC}_{ES_J}I)$. Since J can be started only at $\text{MAX}(^{EC}_{ES_J}I)$, if this is greater than LS_J, there will be an IPD due to positioning of J. However, if the quantity $\text{MAX}(^{EC}_{ES_J}I)$ is less than LS_J, there will be no IPD due to positioning of J. Thus, positioning of J will result in an IPD of

$$\mathrm{MAX}\left[\begin{matrix}0\\ \\ \mathrm{MAX}\begin{pmatrix}EC_I\\ ES_J\end{pmatrix} - LS_J\end{matrix}\right.$$

or equal to

$$\mathrm{MAX}\left[\begin{matrix}0\\ \\ EC_I - LS_J\end{matrix}\right. = \alpha\text{, say.} \tag{26}$$

We have seen that J can only be started at $\mathrm{MAX}\begin{pmatrix}EC_I\\ ES_J\end{pmatrix}$. Then J will be *completed* at $\mathrm{MAX}\begin{pmatrix}EC_I\\ ES_J\end{pmatrix} + d_J$ or $\mathrm{MAX}\begin{pmatrix}EC_I + d_J\\ EC_J\end{pmatrix} = j_1$, say. At this point in time, K can be started if and only if j_1 is greater than or equal to ES_K. If the latter is the case, positioning of K will not increase project duration. But if the former is the case, that is, j_1 is greater than or equal to ES_K, then positioning of K *may* affect the IPD if starting of K is postponed beyond LS_K by an amount $j_1 - LS_K$ which is equal to

$$\mathrm{MAX}\begin{pmatrix}EC_I\\ ES_J\end{pmatrix} + d_J - LS_K$$

So IPD due to positioning of K will be

$$\mathrm{MAX}\begin{bmatrix}0\\ \\ \mathrm{MAX}\begin{pmatrix}EC_I\\ ES_J\end{pmatrix} + d_J - LS_K\end{bmatrix} = \beta\text{, say.} \tag{27}$$

We have seen that K can be started only at time

$$\mathrm{MAX}\begin{bmatrix} \mathrm{MAX}\begin{bmatrix} EC_I + d_J \\ EC_J \end{bmatrix} \\ \\ ES_K \end{bmatrix}$$

K will then be completed at time

$$\mathrm{MAX}\begin{bmatrix} \mathrm{MAX}\begin{bmatrix} EC_I + d_J \\ EC_J \end{bmatrix} \\ \\ ES_K \end{bmatrix} + d_K = k_1, \text{ say.} \qquad (28)$$

At this point in time, L can be started if and only if k_1 is greater than or equal to ES_L. If not, starting of L has to be delayed until ES_L. If latter is the case, positioning of L will not increase project duration. But, if former is the case, that is, k_1 is greater than or equal to ES_L, then positioning of L *may* affect the IPD if starting of L is postponed beyond LS_L by an amount $k_1 - LS_L$ which is the same as

$$\mathrm{MAX}\begin{bmatrix} EC_I + d_J \\ EC_J \\ ES_K \end{bmatrix} + d_K - LS_L$$

So IPD due to positioning of L will be

$$\mathrm{MAX}\begin{bmatrix} 0 \\ \mathrm{MAX}\begin{bmatrix} EC_I + d_J \\ EC_J \\ ES_K \end{bmatrix} + d_K - LS_L \end{bmatrix} = \gamma, \text{ say.} \tag{29}$$

So we have a situation where the positioning of J will tend to increase the project length by α; positioning of K after fixing I and J, will tend to increase the project duration by β; positioning of L after fixing I, J, and K, will tend to increase the project duration by γ. So the net effect of all these four activities I, J, K, L on the over-all project length will be an increase in project duration of $\mathrm{MAX}\begin{Bmatrix}\alpha \\ \beta \\ \gamma\end{Bmatrix}$ which is the same as expression.

$$\mathrm{MAX}\begin{bmatrix} \mathrm{MAX}\begin{bmatrix} 0 \\ EC_I - LS_J \end{bmatrix} \\ \mathrm{MAX}\begin{bmatrix} 0 \\ \mathrm{MAX}\begin{pmatrix} EC_I \\ ES_J \end{pmatrix} + d_J - LS_K \end{bmatrix} \\ \mathrm{MAX}\begin{bmatrix} 0 \\ \mathrm{MAX}\begin{bmatrix} EC_I + d_J \\ EC_J \\ ES_K \end{bmatrix} + d_K - LS_L \end{bmatrix} \end{bmatrix} \tag{30}$$

It may be noted that it is the value of this expression that is obtained in line 18 of the worksheet.

Modified Version of the Above Approach

As it was pointed out in the last chapter, the boundary sum value appears to be a helpful indicator to select an optimal or near-optimal sequence in the case of single nonsimultaneous set of four activities also. Though this does not *always* lead to an optimal solution, it is intuitively felt that the procedure will lead to a *good* solution.

As in the case of three activities, calculations on the basic network are performed and the BS values of the four members of the single nonsimultaneous set are determined. Let these be denoted by BS_A, BS_B, BS_C, BS_D. Then the following guidelines are used.

Step 1: If $BS_A \neq BS_B \neq BS_C \neq BS_D$, do the following. Otherwise, go to step 2.

a) Assign priority in increasing order of the boundary sum values. Then, sequence IJKL will be chosen such that

$$BS_I < BS_J < BS_K < BS_L$$

Step 2: If two or three of the BS values are equal and the quantity is the maximum BS value, then do the following. Otherwise, go to step 3.

a) Construct a network, adding pseudo-precedence requirements such that all the activities having maximum BS, are immediate successors to the rest of the activities in the nonsimultaneous set. Call this network the modified network. Compute the new values of BS, that is BS' of all the activities and assign priority in increasing order of the BS' values.

Step 3: If two or three of the BS values are equal and the quantity is the minimum BS value, then do the following. Otherwise, go to step 4.

a) Construct a network adding pseudo-precedence requirements such that all activities having minimum BS are immediate predecessors to the rest of the activities in the nonsimultaneous set. Call this network, the modified network. Compute the new values of BS that is, BS' of all the activities and assign priority in increasing order of BS'.

Step 4: If two of the BS values in the basic network are equal and this quantity is neither the maximum nor the minimum among all the four BS values, then do the following. Otherwise, go to step 5.

a) Consider the activities having the same BS to be the members of a subset within the nonsimultaneous set. Position the activity with Minimum BS to be first activity in the sequence and the activity with Maximum BS to be the last activity in the sequence.

b) Construct a network by adding pseudo-precedence relationships such that the two members of the subset mentioned in (a) above, immediately 1) precede the activity with maximum BS and 2) succeed the activity with minimum BS.

c) Make a forward pass and a backward pass on this modified network and find out the BS' values of the members of the subset. Position the two activities within the subset such that the BS' of successor is greater than or at least equal to BS' of the predecessor. Thus, all the four activities in the nonsimultaneous set are positioned.

Step 5: If BS values of all the activities are equal, do the following.

a) Position an activity which has maximum LS toward the end of the sequence. Let the other three activities form a subset within the nonsimultaneous set.

b) Add the pseudo-precedence requirements such that all the other three members of the nonsimultaneous set are immediate predecessors to the positioned activity. This will result in a modified network.

c) Make a forward pass and a backward pass on the modified network, and schedule the three activities in increasing order of their BS' values. If, however, there is a tie between two of these BS' values, then, construct one more network adding new additional precedence requirements within the subset mentioned in (a) above. And from the new BS' values, assign the priority in increasing order.

Example Application

Example (1)

Consider the following network.

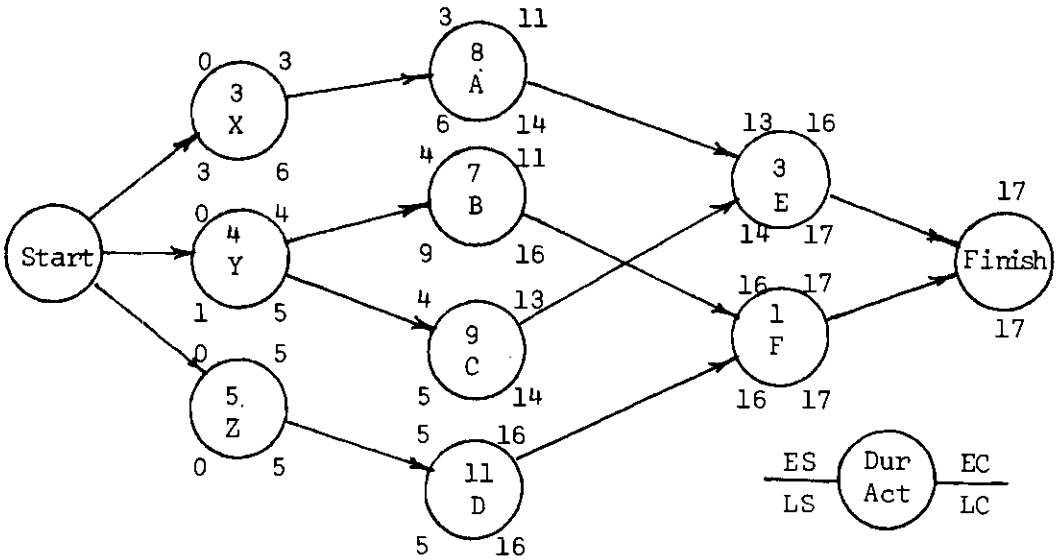


Figure 17. Basic Network Consisting of a Four-Activity Nonsimultaneous Set

Let A, B, C, D be the four members of the single nonsimultaneous set. The forward pass and backward pass are performed on the network, and the BS values of A, B, C, D are determined to be

$$BS_A = 17 \qquad BS_B = 20 \qquad BS_C = 18 \qquad BS_D = 21$$

Step 1: Since $BS_A \neq BS_B \neq BS_C \neq BS_D$, choose the sequence assigning priority in increasing order of BS. Thus, sequence ACBD is chosen as an optimal sequence.

$$(BS_A = 17) < (BS_C = 18) < (BS_B = 20) < (BS_D = 21)$$

By using worksheet No. 2, IPD for each of the 24 sequences can be calculated. The following table gives the IPD for each sequence.

ABCD 22	BACD 23	CABD 23	DABC 26
ABDC 24	BADC 25	CADB 23	DACB 24
ACBD 22	BCAD 23	CBAD 23	DBAC 26
ACDB 22	BCDA 25	CBDA 25	DBCA 26
ADBC 24	BDAC 25	CDAB 23	DCAB 24
ADCB 22	BDCA 25	CDBA 25	DCBA 26

From this, it may be noted that the sequence ACBD chosen as *good* is really optimal. ACBD has the minimum IPD of 22 time units.

Example (2)

Consider the following network

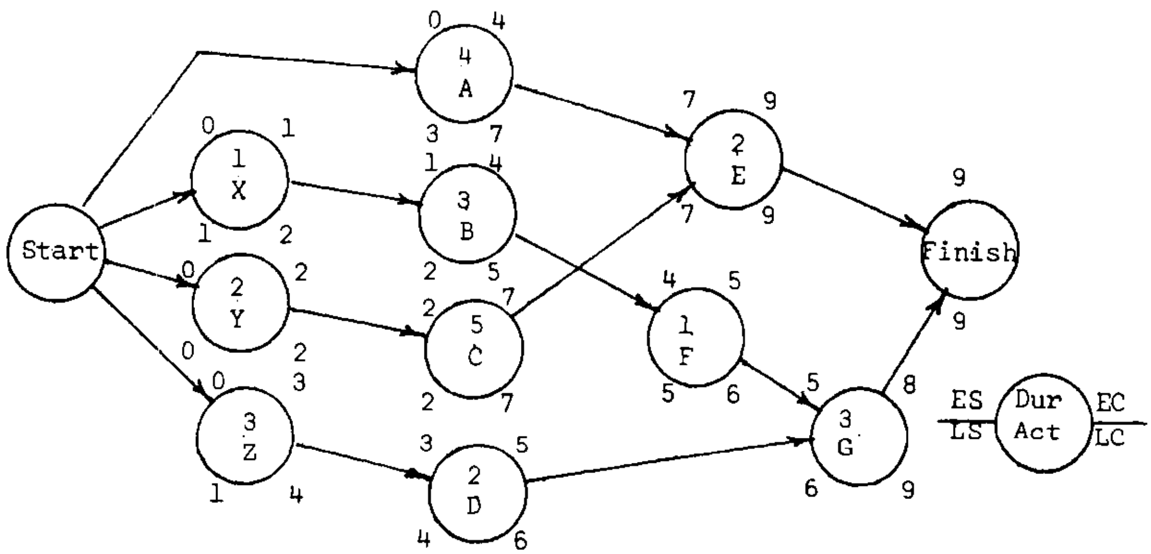


Figure 18. Basic Network--Example 2

Suppose A, B, C, D, are the members of the single nonsimultaneous set. BS_A, BS_B, BS_C, BS_D are determined to be equal to 7, 6, 9, 9, respectively. To find a *good* sequence:

Step 1: Not applicable since $BS_C = BS_D$. Go to step 2.

Step 2: $BS_C = BS_D = 9$ and is the maximum among the BS values of four activities in the nonsimultaneous set. So do the following.

a) Construct a network, adding pseudo-precedence requirements to the basic network such that C and D having maximum BS are immediate successors to the rest of the activities in the nonsimultaneous set, namely A and B. The modified network is as follows.

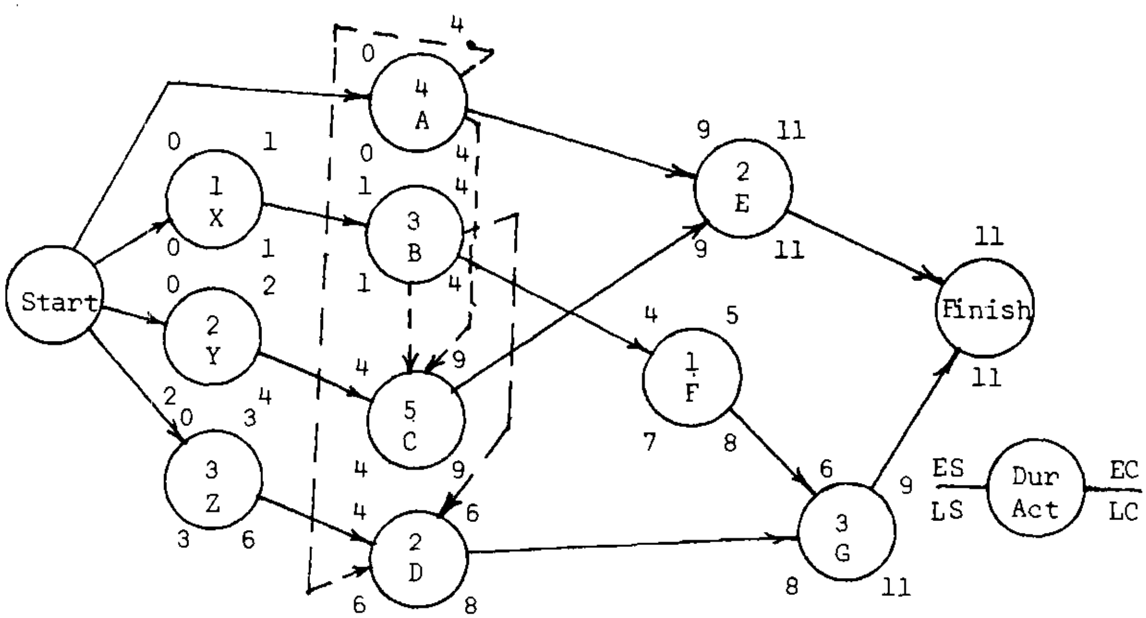


Figure 19. Modified Network With $\{{A \atop B}\} >> \{{C \atop D}\}$

From this modified network, we have $BS'_A = 4$, $BS'_B = 5$, $BS'_C = 13$, $BS'_D = 12$. So choose ABDC; $BS'_A < BS'_B < BS'_D < BS'_C$. The IPD for each of the 24 sequences, calculated using worksheet No. 2, is as follows.

ABCD 8	BACD 9	CABD 10	DABC 10
ABDC 7	BADC 8	CADB 11	DACB 12
ACBD 8	BCAD 9	CBAD 10	DBAC 10
ACDB 9	BCDA 8	CBDA 9	DBCA 10
ADBC 7	BDAC 8	CDAB 11	DCAB 12
ADCB 9	BDCA 8	CDBA 9	DCBA 10

From this, it may be noted that the sequence ABDC, chosen as a *good* one, is really optimal.

Example (3)

Consider the following network.

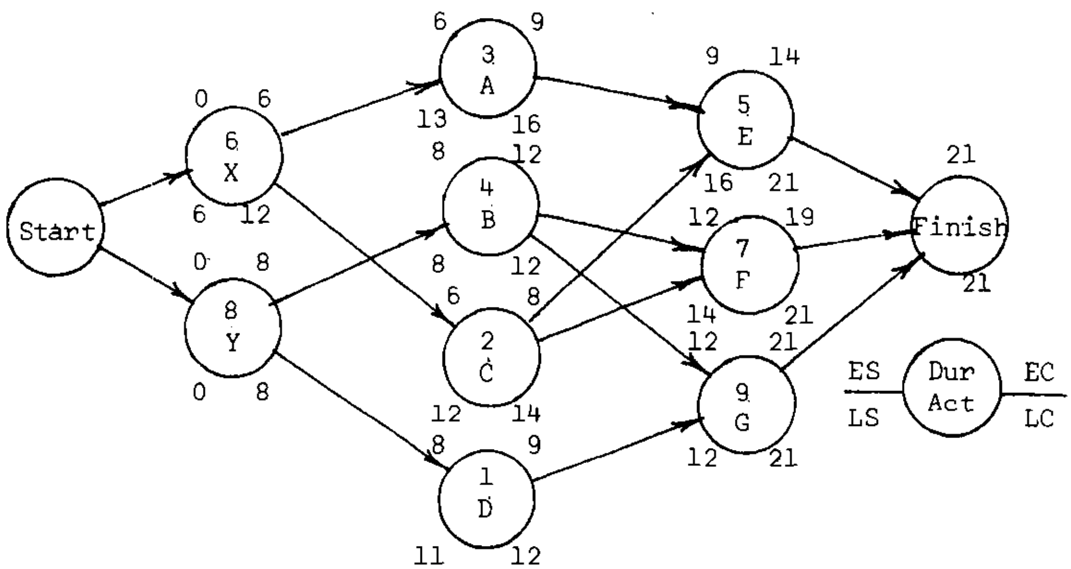


Figure 20. Basic Network--Example 3

Suppose A, B, C, D are the members of the single nonsimultaneous set. Calculations on this basic network give values of

$$BS_A = 22 \qquad BS_B = 20 \qquad BS_C = 20 \qquad BS_D = 20$$

Step 1: Not applicable since $BS_B = BS_C = BS_D$. Go to step 2.

Step 2: Since $BS_B = BS_C = BS_D = 20$ and this quantity is not maximum BS, step 2 is not applicable. Go to step 3.

Step 3: Since $BS_B = BS_C = BS_D = 20$ and 20 is the minimum BS, do the following:

a) Construct a network adding pseudo-precedence such that all activities having minimum BS are immediate predecessors to the rest of the activities in the nonsimultaneous set. That is, make activities B, C, D immediately precede A. A is now positioned. The resulting network will be as shown below.

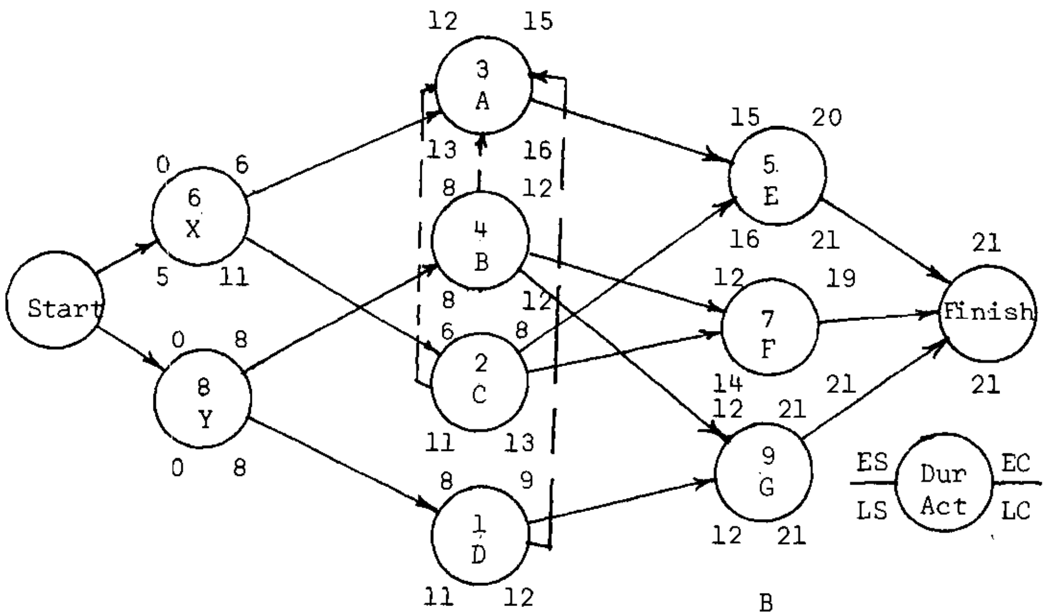


Figure 21. Modified Network with $\begin{Bmatrix} B \\ C \\ D \end{Bmatrix} >> A$

From the above modified network, we have $BS'_B = 20$, $BS'_C = 19$, and $BS'_D = 20$. Since $BS'_C = 19$ is less than BS'_B and BS'_D, C can be now positioned to be at the beginning of the sequence. To determine whether B is to follow D or vice versa, construct one more network by having imposed precedence C >> $\{^B_D\}$ >> A.

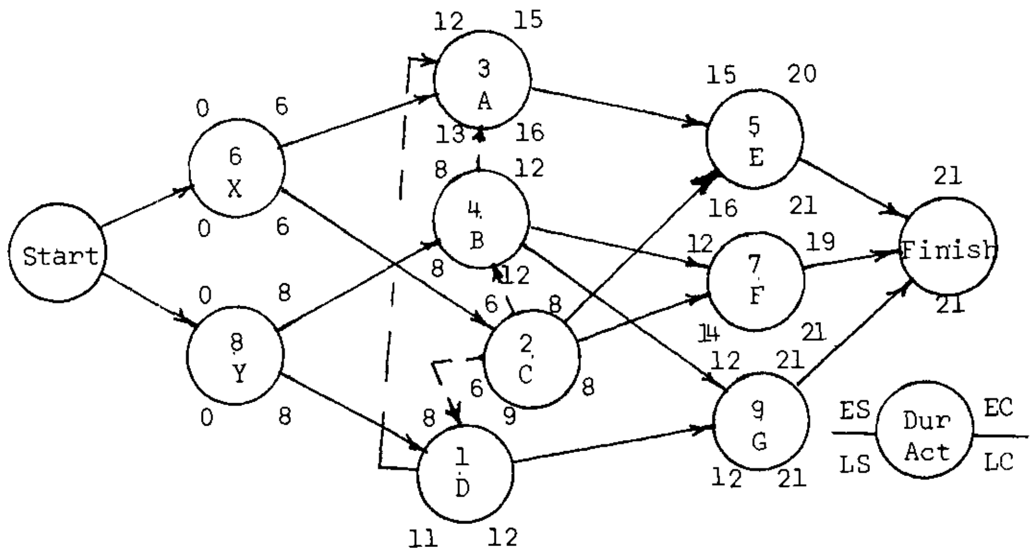


Figure 22. Modified Network with C >> $\{^B_D\}$ >> A

From this, we have $BS'_B = 20$, $BS'_D = 20$. So, B can be scheduled to follow D or D to follow B. Thus, we have CBDA and CDBA as *good* solutions. The IPD for each of the 24 sequences calculated using worksheet No. 2 is as follows.

ABCD 4	BACD 6	CABD 4	DABC 4
ABDC 2	BADC 4	CADB 4	DACB 6
ACBD 4	BCAD 6	CBAD 4	DBAC 4
ACDB 4	BCDA 3	CBDA 1	DBCA 2
ADBC 2	BDAC 4	CDAB 4	DCAB 6
ADCB 4	BDCA 2	CDBA 1	DCBA 3

From this, it may be noted that CBDA and CDBA are the optimal sequences.

For the three examples given, it may be noticed that the sequence that was chosen as *good* on the basis of BS criterion was really optimal. One example will now be presented where the sequence chosen, based on BS criterion, did not result in an optimal solution.

Example (4)

Consider the following network.

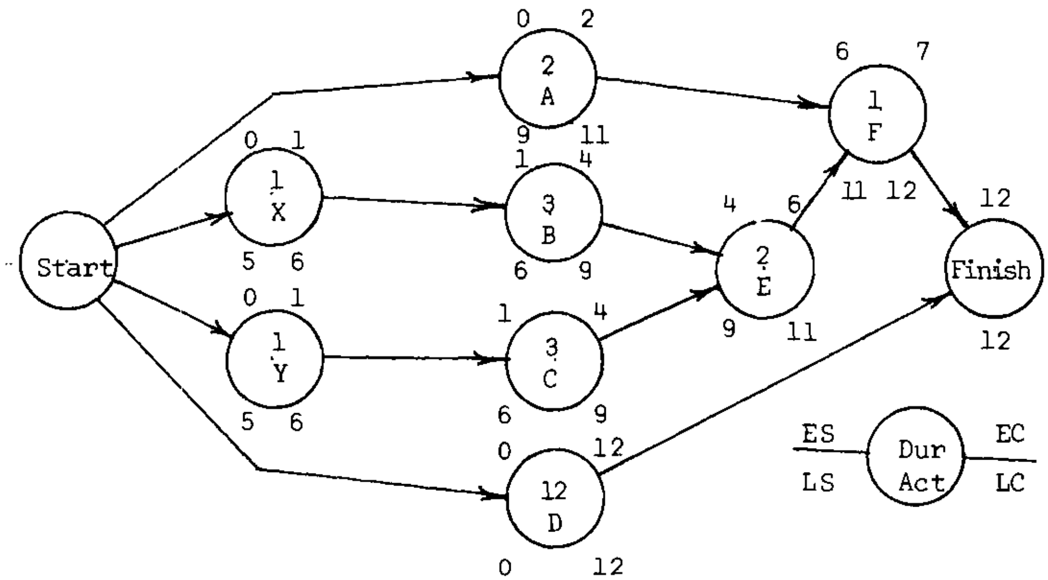


Figure 23. Basic Network for Example 4

Suppose A, B, C, D are the four members of the single nonsimultaneous set. Calculations on this basic network give values of

$$BS_A = 11 \qquad BS_B = 10 \qquad BS_C = 10 \qquad BS_D = 12$$

To find a *good* sequence:

Step 1: Not applicable since $BS_B = BS_C$. Go to step 2.

Step 2: Since $BS_B = BS_C = 10$ and this quantity is not maximum BS, step 2 is not applicable. Go to step 3.

Step 3: Since $BS_B = BS_C = 10$ and 10 is the minimum BS, do the following:

a) Construct a network adding pseudo-precedence such that all the activities having minimum BS are immediate predecessors to the rest of the activities in the nonsimultaneous set. That is, make $\{{B \atop C}\} \gg \{{A \atop D}\}$. Resulting network will be as shown.

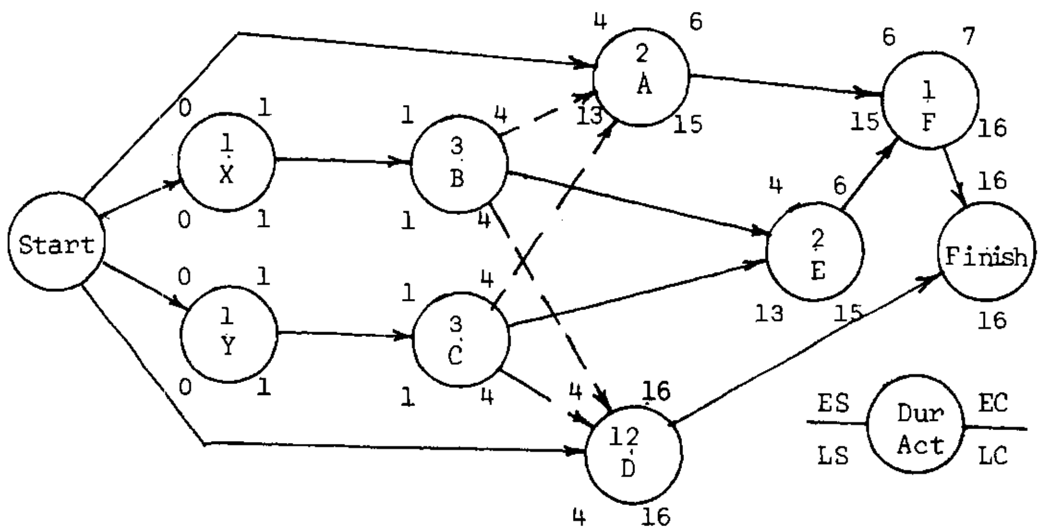


Figure 24. Modified Network for Example 4

From the above modified network, we have

$BS'_A = 19 \qquad BS'_B = 5 \qquad BS'_C = 5 \qquad BS'_D = 20$

So, BCAD and CBAD are chosen as *good* sequences.

The IPD for each of the 24 sequences calculated using worksheet is as follows.

ABCD	8	BACD	9	CABD	9	DABC	11
ABDC	11	BADC	12	CADB	12	DACB	11
ACBD	8	BCAD	9	CBAD	9	DBAC	11
ACDB	11	BCDA	10	CBDA	10	DBCA	9
ADBC	11	BDAC	12	CDAB	12	DCAB	11
ADCB	11	BDCA	10	CDBA	10	DCBA	9

From this, it may be noted that BCAD and CBAD with IPD of 9 units, are not really optimal. However, they are the next best after the sequences ABCD and ACBD with minimum IPD of 8 units.

We have seen that the worksheet approach does give an optimal sequence. The complete enumeration will necessitate constructing 24 different networks and making a forward pass on each of these 24 networks. With the worksheet approach, we are not concerned with any network other than the basic network. The computations are relatively simple. The modified approach of choosing a sequence using *BS* criterion is simpler than the worksheet approach. But we have seen that the selection of optimal sequence is not guaranteed with this modified approach. However, it does tend to indicate near-optimal, if not optimal sequences. The approach using *BS* criterion may be more rewarding when there are more than four activities and if it is felt desirable to find a *good* sequence which is not far from optimal. For nonsimultaneous sets consisting of more than four activities, a worksheet approach to compute IPD for each of the many sequences appears to be tedious and therefore not practicable.

CHAPTER V

LEVELING OF THE RESOURCES AND THE NONSIMULTANEITY CONSTRAINT

The purpose of this chapter is to discuss briefly the *leveling* or *smoothing* of the resources and to indicate the relationship between the *leveling* aspects of basic scheduling problems and the nonsimultaneity constraint. The reasons for leveling and the problems encountered in leveling are described.

Description of the *Leveling* Problem

As indicated in Chapter II on Literature Survey, there are two basic scheduling problems. One deals with *leveling* the demand of resources with a constraint on the total project duration time while the second problem deals with the minimization of the duration of the project with a restriction on the availability of resources. The *leveling* problem arises when it is felt desirable to continue the utilization of resources at a relatively constant rate given that sufficient resources can be procured to carry out the project being scheduled.

Reasons for Leveling

Necessity of leveling resource demands stems from the desirability to a) reduce the peak resource demand or the maximum resource requirement, b) to reduce the amount of time that resources are not fully utilized, and c) to reduce the costs associated with the acquisition and disposal of resources, such as hiring and firing, if manpower is the resource (28).

The difficulties experienced while attempting to smooth the resources are essentially four-fold. These are as follows.

1) The computational magnitude involved in generating all possible schedules and picking the optimum is generally prohibitive.

2) In the case of schedules involving more than one resource, leveling one resource may cause the schedule to reflect lack of uniformity in the demand profile of another resource. That is, the inherent precedence relationships in the project may be such that the requirements for leveling of one resource may be in direct conflict with those for leveling another.

3) Any deviation from the expected activity duration time or from the estimated resource requirements for each activity may necessitate releveling.

4) Lack of progress which results in updating may necessitate releveling.

The literature survey covers some of the leveling procedures which are used to smooth the resource profile. These procedures do not incorporate the nonsimultaneity constraint aspects. Therefore, if a project consists of nonsimultaneous sets of activities, the existing leveling procedures may be of only limited use. It is thought appropriate to bring out some of the general characteristics of situations when the project manager is interested to consider the IPD and *also* the resource requirement profile, while sequencing the activities in the nonsimultaneous set.

The traditional leveling procedures tend to smooth the resources with a constraint on the project duration. Or, within the given project length, activities are scheduled so that this results in as much uniformity of resource requirement as possible. Suppose the resource profile for a small project is as shown in Figure 25.

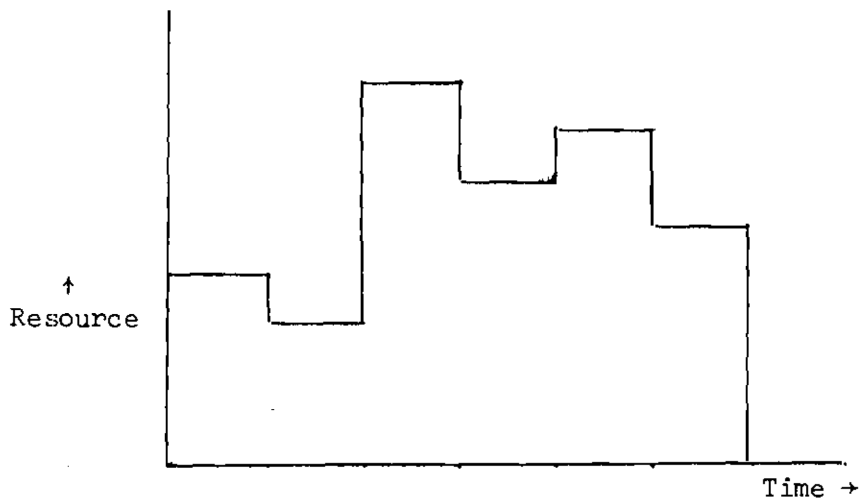


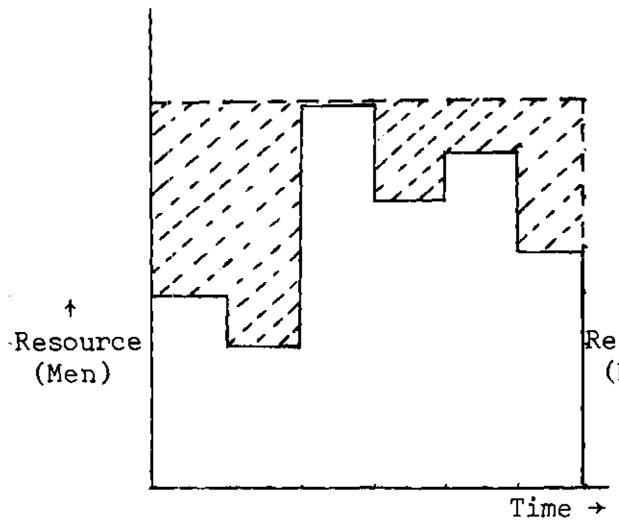
Figure 25. Resource Profile for a Small Project

The management policy may be to

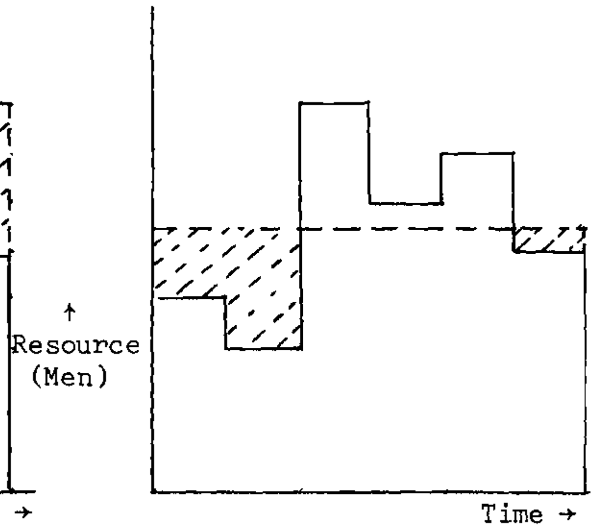
a) keep the maximum resource level always, as indicated in Figure 26 (a);

b) keep an average level and hire extra men during peak demand period, as indicated in Figure 26 (b);

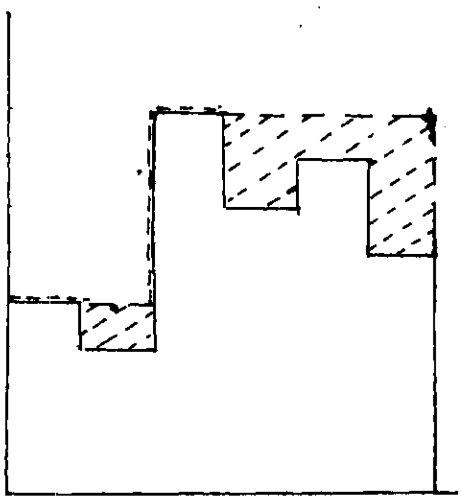
c) hire when required, without firing during the time when the project is in progress, as indicated in Figure 26 (c); or



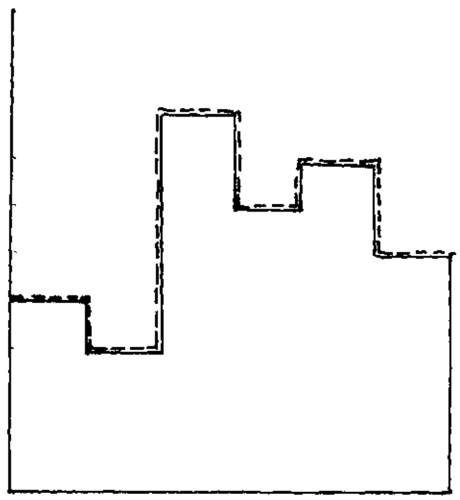
(a) Maintain Maximum Level Throughout the Project



(b) Maintain Average Level Throughout the Project



(c) Hire When Required; Do Not Fire



(d) Hire and Fire as Required

Figure 26. Diagram Illustrating the Resource Profile and the Management Policy

d) hire men when required and fire them when they are not needed, as indicated in Figure 26 (d).

The shaded portion in each of the figures indicates the amount of idle resource-time units. Thus, additional expenditure is incurred due to these idle resources. However, in an effort to minimize this amount of idle resource time, if we follow the policy as indicated in (d) above, then this necessitates frequent hiring and firing which may also result in some additional costs. So, it is evident that the policy will be best decided on the basis of the characteristics of the particular resource profile under consideration. It is also evident that there is a definite cost associated with the variability in the resource requirement. Thus, the cost of alternative schedules will have a bearing on the cost of variability in the resource profile.

The situation is likely to get more complex when there is a nonsimultaneity constraint imposed on one or more sets of activities in the project. We have seen that sequencing of activities within the nonsimultaneous set tends to result in an increase in project duration.

Let us consider a single nonsimultaneous set of three activities with a simultaneity maximum of one. If these three activities are A, B, and C, then we have six possible sequences: ABC, ACB, BAC, BCA, CAB, and CBA. By the worksheet approach presented in Chapter III, it is possible to find out the IPD with each of these six sequences and then choose a sequence that results in a minimum IPD. Thus, with this imposed precedence, we have a certain project length. It is now possible for us to level the resource demands with a constraint on this modified project length. It may so happen that after the leveling operation is done, a resource profile results which is not satisfactory to the project manager. In this situation, the project manager may be inclined to accept a sequence within the nonsimultaneous set resulting in an IPD which is not minimal if such a sequence offers a satisfactory resource profile after leveling. One approach to this interrelated problem involving nonsimultaneity and leveling, may be to determine the IPD for each of the six sequences, then obtain a levelled profile of the resource demands for each of the sequences with a constraint on their respective modified project lengths. Then for each sequence, we have the modified project length and also a resource profile which is *good* from the *leveling* point of view. If we could express the variability of the resource profile in some measure, then it is possible to arrive at a sequence which trades off the loss due to IPD with the gain due to the reduction in variability of the resource profile. The quantitative measures of IPD in time units and variability of resource levels in resource units will assist the project manager in arriving at a feasible schedule which is more economical than other schedules. This approach may be computationally feasible in case of single nonsimultaneous sets having three activities. However, with even four activities, the advantages of this approach may be considerably offset by the computational magnitude in leveling each of the 24 separate project networks. Determination of IPD by the worksheet approach is relatively less tedious; however, finding out the variability in resource demands for each of the 24 schedules could seriously hinder the practicability of the approach. In this situation, it may be worthwhile to limit the number of sequences considered for the determination of resource variability. We may find out the IPD for each of the 24 sequences using the worksheet approach and from these values of IPD, choose the sequences which are optimal and near-optimal. Apply a leveling procedure to each of these selected sequences and then compare the variability and IPD for each of these few selected sequences to arrive at an economically feasible schedule.

Another approach may be to arrive at a schedule without considering the nonsimultaneity constraint such that the resource profile with this schedule is satisfactory from the point of view of uniform demand. Then this schedule can be examined for any conflict of resource requirement due to the concurrent occurrence of two or more members of the nonsimultaneous sets with simultaneity maximum equal to one or more. If this does not happen, then the schedule which was obtained after leveling does not violate the nonsimultaneity constraint. However, if it does, then it is necessary to resolve this restraint. Thus, basically, the approach consists of generating a schedule ignoring the nonsimultaneity constraint, modifying this schedule to level the resource demand, examining it to see whether the resulting schedule violates the nonsimultaneity constraint and then resolving the nonsimultaneity constraint. The resulting schedule may once again be examined for further prospects in arriving at a more desirable resource profile. This approach seems to be intuitively better than the former, in the computational aspects.

The above discussions spotlight the necessity of incorporating the nonsimultaneous constraint aspects into the leveling procedures. Two of the possible approaches are briefly described. No specific procedure is given in this chapter; however, some of the problems that are likely to demand attention while considering the interaction between the nonsimultaneity and the leveling are brought out.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

Since the nonsimultaneity constraint aspects of project management systems have so far received only a very limited attention in the contemporary literature and practice, any useful step in this area will be of considerable significance to the present state of art. Understanding the restrictions imposed by the nonsimultaneity constraint and its characteristics, one is able to appreciate more the impact of the nonsimultaneity constraint on project networks. Optimal solutions to the nonsimultaneity constraint problems are important for three primary reasons. One of them is that it helps to understand and appreciate the significance of choosing an optimal solution. Another reason is that in many practical situations this is of help in minimizing the project length. The third reason may be that it is necessary to be able to arrive at an optimal solution with existing knowledge so that the accuracy of new procedures may be evaluated.

The Worksheet Approach

The solution of nonsimultaneity problems using the worksheet approach is optimal for the following conditions:

1. Single nonsimultaneous sets of three activities having simultaneity maximum equal to one or two.

2. Single nonsimultaneous sets of four activities having simultaneity maximum equal to one.

The worksheet approach presented eliminates the need of constructing separate networks for each sequence within the nonsimultaneous set and thus reduces the computations considerably. The magnitude of saving in computation is of very practical significance. The example networks used herein have contained only a few activities. In practice, the total number of activities may be much more and the computational simplification achieved by the worksheet approach is even more striking. The worksheet approach not only helps in finding the optimum sequence, but also gives us an insight into

a) the effect of a particular sequence on the over-all project length and

b) the effect of additional resource availability on the simultaneous set and sequences within it.

Sequencing within a single nonsimultaneous set of three activities with the worksheet approach is quite easy. With the number of activities increased from three to four, the worksheet approach is slightly more elaborate. However, the relative advantage of worksheet approach even for four activities is considerable, as compared to constructing 24 separate networks and making forward pass on each of these.

Modified Approach Using *Boundary Sum* Values

For the case of single nonsimultaneous sets of three activities with a simultaneity maximum equal to one, the modified approach very definitely indicates a *good* solution which turns out to be an optimal in all the randomly-designed networks tried. However, no proof of optimality has been obtained and therefore it will not be quite justifiable to claim that this approach does always enable one to select an optimal sequence. The modified approach is computationally far easier than even the worksheet approach. In certain cases, we are able to choose an optimal sequence without any additional computations other than the calculations on the basic network.

The relative advantages of the modified approach in case of single nonsimultaneous sets of four activities are even more striking. However, due to lack of substantiating evidence, the approach can only be credited with indicating a *good* solution which may be optimal or not far from the optimal. However, it is justifiable to assume that the complexity of the nonsimultaneity constraint makes any approach based on an intuition have negligible likelihood of always achieving an optimal solution. Though the results do not possess proven optimality, are apparently better than the selection of sequences at random.

Leveling and the Nonsimultaneity

Since the existing leveling procedures do not incorporate the nonsimultaneity constraint, they are of limited use when such a constraint is involved in a project network. The interrelationship between the leveling problem and the nonsimultaneity constraint was briefly discussed. The necessity of focussing attention on both the leveling and the nonsimultaneity constraint was explained.

Recommendations

The research described herein deals with the nonsimultaneity constraint when it exists in the form of single nonsimultaneous sets. It is suggested that useful contribution can be made if the results obtained in this research are extended to cover the situations involving multiple nonsimultaneous sets. The worksheet approach presented to deal with the single nonsimultaneous sets of four activities with simultaneity maximum equal to one can be extended to deal with situations when simultaneity maximum is greater than one.

The most pressing problem in the extension of the approaches described herein is the lack of proof of optimality when they are extended to cover the more complex forms of the nonsimultaneity constraint. The generation of general procedures based on purely analytical techniques seems to be rather intricate due to the extremely complex characteristics of the nonsimultaneity and the existence of the same in many varied forms. Therefore, future research with an objective of developing analytical solution techniques in this area will be challenging.

Another area of future research may be in dealing with situations where resolution of a single nonsimultaneous set generates other nonsimultaneous sets. The chances of this happening are even more when multiple resources are involved. There are also situations where two or more sets of activities exist with a nonsimultaneity constraint on each of these sets and also additional constraint forming another nonsimultaneous set which has some of its members from the previously-mentioned sets. In this case, there will be some activities which are members of more than one nonsimultaneous set.

Sequence Sampling Possibilities

One great difficulty in choosing an optimum sequence lies in the computational magnitude of considering all sequences within the nonsimultaneous sets. If we have seven activities in a single nonsimultaneous set, the number of possible combinations is equal to $7! = 5{,}040$. To consider each of these 5,040 sequences with an idea to find out an optimum is computationally prohibitive. If a small sample of 5,040 sequences were taken at random, then the probability that this sample contains an optimum may not be very large, in general. However, if it is possible for us to position at least some of these seven activities based on a criterion like *boundary sum,* then the number of sequences to be considered will be considerably reduced. Another indicator may be a quantity associated with the boundary sum and the total slack. Thus *boundary sum plus total slack* may be an indicator which gives an insight into sequencing some of the activities. So, if we could determine the partial ordering of desirability, then the number of sequences to be considered will be very much reduced. Even in this case, there does exist some uncertainty as to whether an optimum sequence is there in the selected sample. Investigations may also be undertaken to quantify this inherent uncertainty. Thus, good sampling procedures tend to be of immense help in choosing an optimum.

Computer Application

Since the explicit recognition of the significance of the nonsimultaneity constraint is a recent development, no computer program exists to enable one to choose an optimal sequence. Since the calculations are simple, but enormous and repetitive in nature, the use of the computer may be highly desirable to choose an optimal sequence, especially when there are many networks to be analyzed. It may be possible to develop computer programs to take the basic network data and resolve the nonsimultaneity constraint and at the least, give an optimum sequence. Ideally, it will be very desirable to interface this resolution of nonsimultaneity constraint with resource allocation for all the activities in a network, generating a calendar-dated schedule with a resource loading indication by time-period. The capability of re-examining alternative sequences at each updating should also be built into the computer program package. The advent of remote terminals on a time-sharing basis, enhances the importance of computer processing. The convenience in being able to rapidly foresee the effects of the nonsimultaneity constraint directly affects the project manager's tendency to explicitly identify the nonsimultaneity problems.

Job-Shop Situation

It is suggested that the approaches outlined in this research may be probably adapted to a job-shop situation. The jobs may be considered as activities and the nonsimultaneity constraint may be due to the nonavailability of machines to perform a set of jobs concurrently. It may be possible to sequence these jobs that have conflict in resource availability such that all the jobs are completed in minimum time.

Probabilistic Approach

It was seen that the optimality of a particular sequence depends partly on the estimated duration times of the activities in question. The research herein assumes a deterministic value of the activity durations. A sequence which was determined to be optimal may not remain optimal if there are deviations from the estimated duration times of activities within the nonsimultaneous set or the others. So it may be desirable to consider the uncertainty into the procedures. Thus, the variability of activity duration times may also be investigated in order to arrive at an optimal sequence.

It is highly recommended that research be undertaken to modify the existing leveling procedures so that the inherent nonsimultaneity constraint is fully taken into account and the schedule that results does not violate the restrictions imposed by the nonsimultaneity constraint.

Research Results

This research helps one to understand the restrictions imposed by the nonsimultaneity constraint and its characteristics, thus enabling him to appreciate the impact of the nonsimultaneity problem on project networks. Specific procedures are presented and proven optimal for some of the basic forms of the nonsimultaneity problem. Promising approaches are outlined for resolving more complex forms of the problem.

Though there is much additional work to be done in order to deal with complex forms of nonsimultaneity, this thesis constitutes a significant contribution to advance the existing knowledge and provides guidelines for further research in the area.

BIBLIOGRAPHY

1. Ackoff, Russell L., *Progress in Operations Research, Volume One,* John Wiley and Sons, Inc., New York, 1963.

2. Baumgartner, John S., *Project Management,* Richard D. Irwin, Inc., Homewood, Ill., 1963.

3. Black, O. J., "An Algorithm for Resource Leveling in Project Networks," Unpublished Paper, Department of Industrial Administration, Yale University, May, 1965.

4. Brooks, George H., "Algorithm for Activity Scheduling to Satisfy Constraints on Resources," Published in *Project Management with CPM and PERT,* edited by Joseph J. Moder and Cecil R. Phillips, Reinhold Publishing Corporation, New York, 1964.

5. Burgess, A. R., and Killebrew, J. B., "Variation in Activity Level on a Cyclic Arrow Diagram," *Journal of Industrial Engineering,* March-April, 1962.

6. Christensen, Borge M., "Network Models for Project Scheduling Part 2--Preliminary Scheduling Phase," *Machine Design,* May 24, 1962, pp. 173-177.

7. Davis, Edward W., "Resource Allocation in Project Network Models--A Survey," *Journal of Industrial Engineering,* March-April, 1966, pp. 177-188.

8. Davis, J. Gordon, "The Nonsimultaneity Constraint in Network-Based Project Management Systems," (Doctoral Dissertation), Georgia Institute of Technology, 1967.

9. DeWitte, L., "Manpower Leveling of PERT Networks, *Data Processing for Science/Engineering,* March-April, 1964.

10. Filipetti, George, *Industrial Management in Transition,* Richard D. Irwin, Inc., Homewood, Illinois, 1953.

11. Frambes, Roland, "Next Big Step for PERT," *Aerospace Management,* October, 1961, pp. 77-78.

12. Gantt, Henry L., *Work, Wages and Profits,* The Engineering Magazine Co., New York, 1910.

13. Gutjahr, A. L., and Nemhauser, G. L., "An Algorithm for the Line-Balancing Problem,""*Management Science,* November, 1964.

14. Kelley, J. E., Jr., "The Critical Path Method: Resources Planning and Scheduling," a Paper Presented at the Factory Scheduling Conference, Carnegie Institute of Technology, May 10-12, 1961.

15. Knoeppel, C. E., *Installing Efficiency Methods,* The Engineering Magazine, New York, 1915.

16. Levy, F. K., Thompson, G. S., and Wiest, J. D., "Multi-Ship Multi-Shop Workload Smoothing Program," *Naval Research Logistics Quarterly,* March, 1963.

17. Levy, F. K., Thompson, G. S., and Wiest, J. D., "Mathematical Basis of the Critical Path Method," a Paper Published in *Industrial Scheduling,* Edited by John F. Muth and Gerald L. Thompson, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1963.

18. Levy, F. K., Thompson, G. S., and Wiest, J. D., "Introduction to the Critical-Path Method," a Paper Published in *Industrial Scheduling,* edited by John F. Muth and Gerald L. Thompson, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1963.

19. Malcolm, D. G., "Reliability Maturity Index (RMI)--An Extension of PERT into Reliability Management," *Journal of Industrial Engineering,* January-February, 1963, pp. 3-12.

20. Martino, R. L., *Project Management and Control, Volume III--Allocating and Scheduling Resources,* American Management Association, New York, 1965.

21. Mauchly, John W., "Critical-Path Scheduling," *Chemical Engineering,* April 16, 1962, pp. 139-154.

22. Moder, Joseph J., and Cecil R. Phillips, *Project Management with CPM and PERT,* Reinhold Publishing Corporation, New York, 1964.

23. Moshman, Jack, Jacob Johnson, and Madalyn Larsen, "RAMPS, a Technique for Resource Allocation and Multi-Project Scheduling," *Proceedings, 1963 Spring Joint Computer Conference,* pp. 17-27.

24. Miller, Robert W., *Schedule, Cost, and Profit Control with PERT,* McGraw-Hill Book Company, New York, 1963.

25. Muth, John F., and G. L. Thompson, *Industrial Scheduling,* Prentice-Hall, Inc., Englewood Cliffs, N.J., 1963.

26. Paige, Hilliard W., "How PERT-Cost Helps the General Manager," *Harvard Business Review,* November-December, 1963, pp. 87-95.

27. Shaffer, L. R., J. B. Ritter, and W. L. Meyer, *The Critical Path Method,* McGraw-Hill Book Company, Inc., New York, 1965.

28. Short Course in Project Management with CPM and PERT, sponsored by the School of Industrial Engineering, Notes Compiled by the Department of Continuing Education, Georgia Institute of Technology, March 18-June 3, 1968.

29. Sobczak, Thomas, "Network Planning Techniques," *Research/Development,* May, 1963, pp. 25-29.

30. Taylor, Frederick W., *Scientific Management,* Harper and Brothers, New York, 1947.

31. Verhines, D. R., "Optimum Scheduling of Limited Resources," *Chemical Engineering Progress,* March, 1963, pp. 65-67.

32. Wilson, R. C., "Assembly Line Balancing and Resource Leveling," *University of Michigan Summer Conference, Production and Inventory Control,* 1964.